

UNIVERSITY OF OSLO
Department of Informatics

Filling the Holes with Workarounds: Watching Maps Work in the Terrain

Master thesis

Wilhelm Arthur Sandberg Damsleth

Spring 2013



Filling the Holes with Workarounds: Watching Maps Work in the Terrain

Master thesis

Wilhelm Arthur Sandberg Damsleth

© Wilhelm Arthur Sandberg Damsleth

2013

Filling the Holes with Workarounds: Watching Maps Work in the Terrain

DAMSLETH, Wilhelm Arthur Sandberg

wadamsle@ifi.uio.no

<http://www.duo.uio.no>

Abstract

The actual, situated use of a computer system differs from the use planned by the developers in a range of ways. This masters thesis explores some of the ways in which situated use takes place. Using grounded theory methodology, a series of interviews conducted at a large Norwegian government organization is analyzed, and then complemented with quantitative data from an opinion poll. The thesis identifies several ways of using the system, building on and extending Gasser's (1986) notion of workarounds and Lévi-Strauss' bricolage. The thesis further develops a set of characteristics for these constructs, and argues that this presents a useful vocabulary in discussing how situated use of a formal system actually takes place. The thesis concludes that situated use necessarily has to differ from the planned use of a system, and that this is a generative benefit that developers can learn from and facilitate by improving the system to better support situated use.

Preface

This thesis represents not only the culmination of work and studies for my master's degree in *Informatics: Design, Use, Interaction*, but it is also the climax of five incredible years at the Department of Informatics, University of Oslo.

I am exceptionally thankful for the skilled guidance and insightful advice given to me by my thesis supervisor Professor Tone Bratteteig. Tone is a calm and appreciative helmswoman, and I cannot help but feel that we both share a deep enthusiasm for the subject, as apparent in our discussions. Additionally, I owe many thanks to Tone for bringing me on as a teacher's assistant in her masters-level course on Computer Supported Cooperative Work – a truly wonderful experience that has given me a wealth of insight, as I have given it away.

Acknowledging my progress with the methodology in this thesis, Tone and Sisse Finken gave me the opportunity to guest lecture on Grounded Theory in Sisse's masters course on Qualitative Research Methods, an experience I am very grateful to have.

I am very thankful for Professor Ina Wagner's thorough feedback on my work, and her help in the development of some of the key concepts through our discussion.

I would also like to thank those whom I interacted with at the University of California, Berkeley, who perhaps unbeknownst to them, introduced me to the sociology of work and helped me develop my own views and standpoints.

I am further grateful for the time and flexibility my employer has provided me with to complete this thesis.

I cannot overstate how important the friends I have made have become to be, and I value them ever so much – especially those of the “Dream Team”, who truly live up to their name.

I believe that the realization of this thesis has been the single biggest undertaking I have ever set upon in my life. I realize now, at the close, that I could never have done so alone. I would like to present the most special thanks to my parents for their continuous support and care. The biggest recognition goes to my beloved Monika, whose support, attention, motivation, happiness and love I could never have made it without. Thank you.

Oslo, April 2013

Wilhelm Damsleth

Table of Contents

Abstract	V
Preface.....	VII
1. Introduction	1
1.1 Workarounds, Computing Slip and the Bricolage	4
2. Methods and Subjects.....	9
2.1 Grounded Theory	9
2.1.1 Data Collection.....	11
2.1.2 Coding – Open and Axial Coding, and Theory in Memos	14
2.1.3 Programmatically Analysing and Visualizing the Data	20
2.1.4 Integration and Relation	25
2.2 Organization and System	28
2.3 Subjects and Selection	30
3. Findings and Discussion.....	33
3.1 Process	33
3.2 Embodiments of Workarounds.....	36
3.2.1 Communication.....	37
3.2.2 Cheating the System – Working Around the PO Types.....	38
3.2.3 Keeping Track and Restructuring	40
3.2.4 Bridging to the Formal	46
3.2.5 The Good Intentions of Disservice and Disregarded Feedback.....	47
3.3 Characteristics of Workarounds and Bricolages	51
3.3.1 Misfit	51
3.3.2 Transparency.....	52
3.3.3 Peripherality.....	55
3.3.4 Detective Work and Constructed Rigidity	60
3.3.5 Trusting the Bricolage	62
3.4 Accountability and Organizational Ordering.....	64
3.5 Workarounds, Bricolages and their Characteristics	71
4. Contrasting with Quantitative Data: An Opinion Poll	81
4.1 Acknowledging the Opaque Workarounds	89
5. Summary and Discussion.....	91
6. Conclusion	101
6.1 Further Research.....	106
References.....	109
List of Figures.....	113
List of Tables.....	115
Attachments	117

1. Introduction

When is a computer system used as planned? If the developer of a computer system could envision all possible use cases, what do we really need the human operators for? Why do we put humans, the most random of beings, in front of the machines to operate them? And what happens when the features required by the human diverges from the possibilities provided by the computer system?

In this thesis, a study of real-world usage of a system seeks to uncover how the system is or is not used according to its intentions through interviews with its users. Specifically, the research focuses on how the users have diverted from planned use of the system and rather employed it in novel ways, for example by creating their own alternate systems or by modifying variables to suit their desires. Central to the thesis is the concept of workarounds, as described by Gasser (1986) and others, and how they juxtapose with the planned use of the system. The thesis seeks to show that the study of computer systems with a focus on workarounds can provide interesting results and guidance for future development of the system, and that the concepts relating to how situated use differs from planned use, among which we find workarounds, can be nuanced through a typology and a set of characteristics.

The study seeks to understand the situated use (Suchman 2007) of a formal system, encompassing users, organization and technology together. Central to the thesis is the belief that the study of how situated use of a system drifts away from planned use will bring about interesting ideas and insights, and we will see how situated action is unique and worthy of individual study.

In essence, the purpose of the study is given in the research question:

What are some of the ways in which situated use differs from planned use, and how can other ways of working with the system be understood and described?

The thesis holds that deviations from the planned order of work, among which we find workarounds, are not necessarily undesirable or bad in themselves. If work could be undertaken entirely according to script, then indeed even the need for a human operator of the system would become questioned, perhaps resulting in an underutilization of both human and computer.

Therefore, the thesis sees workarounds as interesting opportunities for study and heightened understanding of situated use. Situated use rarely takes place without some form of workarounds, and as such, the development of a language to talk about workarounds is seen as a contribution worthwhile to make.

Given this need of a vocabulary of divergence between planned and situated use, one of the main desires of an outcome of the study is the construction of a typology and a set of characteristics

of workarounds. Hopefully, such a typology can be extended, reconsidered and appropriated by other scholars in the field; nevertheless, it is conceivably a beginning. A typology can also contribute to the development of shared attributes across the workarounds represented by the typology, leading to the discovery of common traits, which can help further research and development.

The research will draw on concepts from the studies of CSCW, or computer supported cooperative work-studies, combined with information systems research and to some extent the sociology of work, in the perspective of the Scandinavian tradition on system development.

As a study of the situated use of a computer system where multiple users carry out interrelated but different tasks, the research is positioned within the field CSCW. The CSCW field offers an excellent library of concepts and does not discriminate against perspectives from the sciences of technology or sociology, which this research is positioned within. Especially, the thesis draws on the notion that the system offers what is called a *common information space* (Schmidt and Bannon 1992), in that it represents, structures and transforms the data it conveys between a multitude of actors with both common and differing tasks in various task chains. The system is entirely dependent on the input from the users, and as it does not create any data on its own, it is reasonable to interpret it as a shared information space.

Further, as we shall see, the formal system does not always accommodate and support cooperation to the extent required by the users. The tools and constructs that the users create attempt to make up for this shortcoming, since the users have a need for cooperation not satisfied by the system. As such, a positioning within the field of CSCW becomes necessary for the interpretation of all of the findings.

In interviewing end users, the study takes a bottom-up approach to systems development not unlike that of the Scandinavian tradition (Bratteteig 2004). This approach, which is commonly characterized by the belief in the users being the best source of insight into their field and therefore encouraging close collaboration between users and developer, can also be applied in the post-implementation understanding of a system, as it will be in this case. Further, the tradition and its practitioners are “*more inclined to appreciate situated knowledge and local action*” (Bratteteig 2004, 19), making for an excellent opportunity for the study of situated use.

The case questions the use of an Enterprise Resource Planning, or ERP, system in a larger Norwegian government organization. The ERP system is an administrative system of considerable size, intending to support many aspects of management, such as that of human resources, accounting, purchasing, logistics, resource planning etc. Intended users of the system are assigned one or more roles that give them access to certain areas of the system, where they can view, structure and submit data to the central database.

Subjects representing different departments of the organization were interviewed, and the interviews were then subjected to grounded theory analysis, giving results that are both inferable directly from the text as well as insight beyond what the interviews in themselves uncovered. The subjects that were interviewed are either purchasers or financial controllers, in addition to one subject who is an instructor, coming from a purchasing background. This selection was made to ensure comparability between the interviews, so that multiple workers can shed light on the same range of issues in different parts of the organization. It is possible, perhaps even likely, that a wholly different set of results would come about if the study had focused on other fields of work like human resources or perhaps logistics.

The interviews were conducted within a context of and analysed with grounded theory principles. A detailed introduction to the grounded theory methodology, along with an introduction to the subjects and the organization, will be given in chapter 2.

Interesting discoveries included the invention of novel and local techniques for problem solving and situational awareness. The findings are both varied and colourful, and they will be detailed in chapter 3. The chapter will begin with a description of the purchasing process in general, along with comparisons of how it is actually performed in some cases, before moving on to the presentation and discussion of particular results. To present the most tangible results first, some of the concrete embodiments of workarounds that were found will be presented and discussed, after which we will entertain a discussion on the characteristics of workarounds in a more general perspective. The thesis will then present a discussion on workarounds, accountability and organizational ordering, before ending with the presentation of a typology of workarounds and their characteristics. Some of the theory that forms a major backdrop to the results will be presented later in this introduction, while some other theory will be introduced in the discussion.

Further, the study was fortunate enough to be granted access to the quantitative results of an opinion poll that was conducted in the organization during the beginning of this research. This data has been analysed in the hopes of uncovering both similarities and contradictions between it and the other results provided in the thesis, in order to interpret the results of the main research as indicative of other areas as well. The analysis shows interesting correlations between self-evaluated competence and the prevalence of workarounds, together with evidence for fairly deliberate wishes for working around. This analysis, along with some discussion, is in chapter 4.

In closing, the thesis summarises the findings before drawing conclusions and provides some suggestions for extensions of the research done herein, as we shall see in chapters 5 and 6. But first, some background on workarounds and the difference between map and territory.

1.1 Workarounds, Computing Slip and the Bricolage

While some theory will be pulled in during the discussion, a background of common concepts is necessary to position the research and give an overview that the ideas presented later will build upon.

Essential to this paper is the notion of the workaround. In his renown paper Gasser (1986) presents an interpretation of situated computer use that, even though coined before the establishment of the field, has become central to CSCW research. His explorations of workarounds and their related phenomena are important to the understanding of actual situated use of computer systems. Especially, three types of adaption work are presented: The work of fitting, augmenting and working around computing (Gasser 1986, 214) – all descriptive of situations where computing is adapted or used in an unintended way to achieve some other effect. As Gasser clearly states, these phenomena are essential to most computing work, and should they be hindered, business operations would degrade considerably fairly rapidly. Gasser further argues that the carrying out of these sorts of adaption are also work, and that by extension, they can be interpreted in the same ways as primary work would be.

Gasser also introduces the concept of “*computing slip*” (1986, 212), which are temporal or permanent situations where there exists a misfit between the work that the computer system is supposed to support and the work that is being attempted. Within this concept are the three major categories of *fitting*, *augmenting* and *working around*. In short, fitting work is work that is carried out in order to adapt the computer system in some way to the organization or the work being carried out, for example by making adjustments to the computer system or adapting the organization to better fit the computer system as it is. Augmenting work is work that is performed in addition to the primary work in order to make up for the misfit of computing slip. As opposed to fitting, augmenting work provides no permanent solution and must be constantly carried out in order to keep achieving the benefits. Finally, working around is the practice of using a computer system for another purpose than intended, for example by manipulating the data entry, or not using the computer system at all while relying on some other system.

These concepts are all highly relevant in the study of computer supported work, as they allow for explanation of the phenomena that occur when the intention and capabilities of the system does not line up with the desires and requirements of the end user. Critically, the study of computing slip allows for the identification of such mismatches along with the possibility of redesigning or improving the CSCW arrangements, which as argued is to be considered the main purpose of this field of research (Schmidt and Bannon 1992). As such, the relevance for this thesis

lies in the vocabularies of the workarounds and their ability to explain and bridge the gaps between planned and situated use of the computer systems that are introduced into an organization.

Another important feature to note at this time is that the definitions of the workarounds are all problem-oriented, in the sense that they all presuppose a flaw or problem in the implementation of the system that prevents it from reaching its highest levels of acceptance and efficiency. That is, the workaround in itself carries a negative connotation, or brings with it an air of undesirability or a suboptimal configuration. It is the intention of this thesis to discuss nuances or variations on the subject of workarounds that might show that the workaround in itself might not be undesirable or unwanted, or even more crucially, that the workarounds and the planned use of a system can be so integrated that it can be close to impossible to draw the line – and that the placement of the line between a workaround and planned use of a system changes based on the observers point of view.

A study focused entirely on breakdowns, on the other hand, could become a "*controversy study*" (Ribes and Lee 2010, 238). Such research might be well suited to uncover organizational tensions, conflicts of interest or other pessimistically asserted situations. Breakdown studies can be applied both on smaller scales locally and wider scales organizationally, but since such a study is more problem-oriented, a positivistic study of opportunistic workarounds seems better suited for the discovery of actual situated use and invention.

This is not to challenge the role of the formal system; when one might ask, then which is the authoritative source of data? Is the workaround the official version, "*reinstalling the informal in its privileged position to overcome the limitations of the formal*" (Berg 1997, 151)? We do not have to observe the workaround as an acknowledgement that the formal system is now delegated as a subordinate of the workaround; rather, it can be seen as an attempt to make do or make better with what is at hand and the tools that are given; either through observed limitations in the tools functions, from the lack of skill in employing the tools or for other reasons that we will explore. A focus on how the workers bridge the gap between what is offered by the formal system and the requirements of the real world becomes appropriate.

While allowing us to see the practices of working around as a dimension of work itself, the adaptation work presented by Gasser then becomes juxtaposed with some notion of what the proper work is, and that Gasser's terms and work without workarounds thereby come to carry some internal dissent as to what work should actually take place. There is room, then, to nuance and extend these concepts, and explore where on the scale between planned use, and the supplementary adaptation work, actual situated practice can be placed.

In a discussion on how to overcome such limits of formal systems, Berg (1997) elects to talk about the activity of *tinkering*, and how the human workers use their available skills and tools for

immediate problem-solving. He claims that the workers are not competing with the formal systems, and that the tinkering activities are natural components of the unfolding of work. The world represented by the formal system is not an incorrect or misleading one, and as such, one cannot talk of appropriation work as this assumes that the system does not carry the same intentions as the workers. In particular, Berg holds that the power of the formal tools directly come from their ability to not carry forth the current state of the entire world, and that the fact that the world is underrepresented by the formal tool is what creates powerful combinations of systems and workers. He states that *“the generative power of formal tools, then, lies in the very existence of the gap between the work practice and its formal representation”* (Berg 1997, 153), acknowledging that it is the very variations between what the system offers and what the users desire that enables the worker to provide added value to the flow of work. This view positions the adaptation work as not only part of the primary work, but perhaps more so than the work that is not the tinkering activities. In this light, adaptation work can be interpreted as more rewarding to work than the actual planned work.

Should we choose to empower the worker even further, and add generative powers to tinkering activities described by Berg, we arrive at *bricolage* as pioneered by Lévi-Strauss (1966). A bricoleur, by his definition, is a person adept at going about a task by composing or arranging at-hand tools and artifacts in a way different from the way a professional designer of such a tool might have done it, while still achieving more or less the desired results. The key difference between the pure professional and the bricoleur is that the latter employs the resources available in the immediate vicinity, always both constricted and empowered by this scheme.

“The 'bricoleur' is adept at performing a large number of diverse tasks; but, unlike the engineer, he does not subordinate each of them to the availability of raw materials and tools conceived and procured for the purpose of the project. His universe of instruments is closed and the rules of his game are always to make do with 'whatever is at hand', that is to say with a set of tools and materials which is always finite and is also heterogeneous because what it contains bears no relation to the current project, or indeed to any particular project [...]" (Lévi-Strauss 1966, 17)

The product of the bricoleur, then, will be the *bricolage* – an arrangement set to serve some purpose that is both immediate and close, yet deserves a solution more than a remedy. As we will see later, the concept of the *bricolage* is relevant in the study of workarounds because those who are engaged in the act of working around often do so in a way similar to that of the bricoleur, using tools and materials at hand rather than the formal and structured tools of the computer system.

The bricolage can be seen as the extension of the applied tinkerer, where the worker not only seeks to solve whatever challenge is at hand with what is presented, but in doing so, creates something new. Bricolage in the context of groupware and CSCW has been analyzed by others, and for example Wynn (1997) finds that the ability to create in this fashion empowers and enables tools that can be distributed and become accepted for wider use in an organization. Since the bricoleur's creations in the digital domain are suited for distribution and replication, it becomes applicable to talk of not only the act of bricolage, but also bricolage as an artifact. This thesis therefore extends the concept by describing the bricolage as a noun – bricolage is the product of the bricoleur.

As opposed to being subjected to the will of the developer, the bricoleur is capable of designing her own tools that mediate and bridge the formal and the actual work. Barley and Orr (1997) describe the emergence of a new kind of job position, that of the engineer-technician, who is skilled at the primary tasks of the trade but simultaneously also capable of creating and modifying the tools necessary to perform. They argue that traditionally, the developers and the users of a given system have been separated into different functional and organizational structures: It is the purpose of the user to be an expert in the understanding and execution of their field of work, and it is the job of the developer to be an expert in creating tools to support work.

This pattern is changing, in that the users are gaining more knowledge about the development, features and possibilities of computer systems in general, and that the increased embeddedness of developers cause them to have greater insight into the business processes and operations of any given field of expertise. This is further underlined by the observed organizational change, which is showing signs of developers being located closer to the users and some users being appointed development experts responsible for communicating specific needs to developers. These roles are merging into that of the engineer-technician, who has great knowledge of both the field of work and the adaption of the tools at hand. The work of creating and adapting the tools might then become embedded in the job itself, as the accounting professional takes on more of the jobs of the developers and integrators in their development of their own workarounds.

This is also how the workers undertake their own *textualization*, a concept described by Zuboff (1988). In textualization, the practices and routines of the work is put down in codified media, like text or computer code, to allow for the automation and routinization of work – it is the act of identifying how and what can be represented digitally, and in happening, it represents the electronic interpretation of the work itself. Textualization can be empowering, in that it allows for new and revolutionary ways of structuring and working with data, but it can also be restrictive in how the end users of a system will be subjected to the textualization performed by another party. The process can impose a certain order of work onto the system, which the end users have to abide by. In the

setting of the emerging engineer-technician, though, the act of textualizing can be performed by the workers themselves, and not by the system developers or consultants. This ability becomes a central notion of this thesis, in that we will observe the emergence of a shift in who performs the textualization of work from developers and system providers to how the end users adapt and create in their situated work.

As follows, central to this thesis is the belief that it is not exclusively the responsibility of the organization to adapt to the groupware when new technology is introduced in a workplace setting. Indeed, it is often presumed that it is only humans and organizations that have the ability to adapt, and not the system. When Ciborra (1997) asks in the title, "*What does Groupware Mean for the Organization Hosting it?*", there exists an assumption that it will be up to the organization to change in order to accept the new system into their midst, while it is not asked how we can modify or adapt the system to better fit the desired or actual situated work taking place.

The thesis becomes a response to the sixth challenge posed by Jonathan Grudin (1994), which states that "*the almost insurmountable obstacles to meaningful, generalizable analysis and evaluation of groupware prevent us from learning from experience*" (Grudin 1994, 97). This calls for the development of a vocabulary to discuss situated use, which will be presented in this thesis.

Positioned in CSCW, sandwiched in-between the technicalities of Information Systems studies and the product-oriented Human-Computer Interaction field (Grudin 1988), this study seeks to uncover what forms workarounds can take and how we can better understand them. But before we dig into the discussion of the findings, let us take a look at how the study was conducted.

2. Methods and Subjects

The choice of methodology and its methods in many ways affects the structure, tonality and results of a study. In this thesis, the desire to get quickly out into the field to begin preliminary analysis combined with the possibilities given through the openness and width of the research being conducted naturally led to the choice of grounded theory as methodology and analytical approach. Grounded theory is especially suited to this form of study since it allows for an open-ended and interpretivist approach, allowing the data and its analysis to continuously feed the process, never restricting or shunning particular areas or problems. Further, the methods scale well vertically for both low and high numbers of data sources, as well as horizontally depending on how in-depth one wishes to get.

Other quantitative, ethnographic methodologies and methods might also have been suitable for the study. For example, one could envision the study leaning a bit further towards a study of the situated uses of the tools being discussed, conducted as participatory observations. As argued by Crang and Cook (2007), such an approach could be easily extended with focus groups and perhaps participatory design sessions, taking the study much further into the prescriptive areas of development and action research (Checkland and Holwell 1998). In the realm of CSCW, though, this might have been a desirable direction for further research – *“enter, and you must change”* (Schmidt and Bannon 1992, 5).

In addition to the grounded theory that is presented here, some quantitative analysis of an opinion poll was also undertaken, which will be presented and discussed in chapter 4. The poll was not created as part of this research, but access to the data was given, and its value will be that of either supporting or contradicting the other results presented in this thesis.

This chapter will detail how the data was collected, analyzed and how concepts were extracted from the data. We will thereafter look at the organization being studied, and in closing get a brief introduction of all the participants who were interviewed for the study. But first, let us get an introduction to grounded theory.

2.1 Grounded Theory

Grounded theory is a qualitative research approach that seeks to develop theory from a set of data, as opposed to other scientific methodologies where one for example might develop a theory and then collect data to test it against. As the name suggests, the aim of grounded theory is to develop several theories that are grounded in the data being analyzed. Grounded theory is open-ended and does not discriminate against any possible result, but is fairly work- and data-intensive, requiring the

constant and repeated analysis of the collected data. One of the key benefits of grounded theory, though, is that it constantly ensures that the theory that is being developed is grounded in the data, and thereby usually ensures the relevance and truthfulness of most of the findings, albeit not usability.

Extremely briefly, grounded theory is applied by transcribing a set of source data, typically interviews, then applying codes, akin to categories, to different parts of the transcribed texts followed by the grouping of these into axial codes. The process is repeated multiple times on the same data, and furthermore with new data. The structure of the resulting set will hint at various correlations that will then form the basis of the theory – a theory that is grounded in the data.

Grounded theory might seem intimidating and hard to grasp at first, but after some consideration, scholars can come to appreciate the immense insights gathered through the application of this process, and the ease of which one can work with the data after familiarizing oneself with the methods. As happened in this research, a point came where the internal relationships of the methodology opened up and understanding was attained, and the sudden onset of usefulness of these methods could truly be appreciated. Bordering on a promise; learning grounded theory can be particularly rewarding, and is certainly a highly interesting skill in itself.

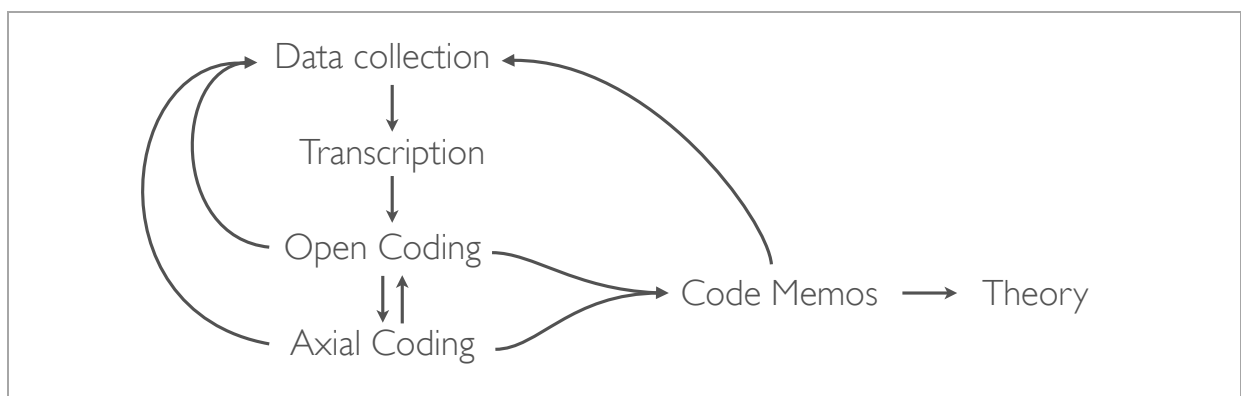


Figure 2.1 Phases of grounded theory as applied in this study. The data was collected through semi-structured interviews.

The process is mostly based on the version of grounded theory (or GT) put forth by Strauss (1987). However, as Strauss remarks, no two grounded theory processes are the same and they will always be subject to the needs of the individual researcher. As such, Strauss does not give a definitive, prescriptive guide on grounded theory, but rather a series of examples and suggestions that the researcher then is free to pick from at their own will. A breakdown of the key phases of a grounded theory process as applied in this thesis is shown in Figure 2.1 – a personal interpretation and adaptation of the methodology, but still largely very representable of the method and its processes.

The approach taken in this study can be seen as fairly interpretivist (Silverman 1998) as it looks mostly at textual data generated from interview sessions, as opposed to more post-modernistic approaches, where one for example might apply grounded theory by coding visual data or other sources. The use of semi-structured or open-ended interviews is also typical of the interpretivist paradigm.

It is important to point out that grounded theory is an iterative process, meaning that each step of the process will again contribute to changing the process when it is repeated again. Iteration through repetition is a key part of GT; each step of the process will feed the next step of the process, and once one iteration has been completed, the results from that iteration is again influencing how the next iteration will be performed. This continuous adaptation and repetition should enable the researcher to extract an abundance of data from the source, and it is through the production of a mass of data one can start to extract key concepts and relationships that eventually will turn into theory.

If grounded theory is such an open-ended and open-minded methodology, then how does one direct the research into the desired area of interest, so as to produce results that are applicable to the study at hand? The researcher's first and foremost tool in ensuring the relevancy of the collected data is by guiding the data collection accordingly. As in all scientific processes, if the collected data is worthless, no amount of processing or analysis will yield an interesting or usable result. It is up to the researcher, then, to ensure that the output of the data collection is relevant, but without imposing bias or prejudice on the subjects being analyzed. A most challenging process, yes, but crucial and principal of all scientific research. In this study, semi-structured interviews were performed, and the planning and execution of these will be discussed shortly. Further, after the collection of data, the researcher has the ability to ensure the relevance of the collected data through the application of codes in both the open coding and axial coding phases. Here as well the process is open to guidance and direction of the researcher, which will be discussed further after the presentation of the data collection phases.

Given this short introduction, we will now walk through the grounded theory process as performed in this study. Some of the details on grounded theory presented here were presented by the author at a guest lecture on grounded theory in the course Qualitative Research Methods at the Department of Informatics, University of Oslo.

2.1.1 Data Collection

As with any methodology, data collection forms a crucial part of grounded theory – no amount of processing can recover bad or non-existent data. We will now take a look at how data collection through semi-structured interviews is optimized for use in a grounded theory process, and how

grounded theory also can be applied to other kinds of data, such as audio-visual data. An overview of sources and their applicability in GT is shown in Table 2.1.

In principle, grounded theory does not discriminate against what kind of data is being used, but the data must always be transcribed or converted to a format that is suitable for easy coding. The central requirement of any data source that is to be analysed with GT is that it is structured in such a way that it is codable, homogenous, and comparable.

The first requirement, codability, is met by making sure that the data exists in a format where it can easily have quantitative codes applied to it, and thereafter the codes can be extracted for analysis. For example, a text that exists digitally in an editable format is a prime example of a very codable piece of data. Texts that exist on paper are equally codable, but will be more time-consuming to extract coded data from, requiring a lot of manual labour on the part of the analyser. Other kinds of data can also be used, such as audio or video. If the audio is speech, it can be transcribed, turning it into a very suitable source for coding. Similarly, if the video is of an interview or a similar situation where dialogue is the primary content, it will be suitable for transcription and analysis. Video and audio from conversations have the additional benefit of carrying more than just the content of the words; these sources can convey nuances such as tonalities, emotions, aggression, enthusiasm, rate of speech etc. These data points can be transcribed together with the conversation itself, making for interesting analysis – why are the subjects angry when they talk about these issues, or why are they enthusiastic about those topics? On the other hand, video and audio can be very challenging – but simultaneously extremely successful – if they record other kinds of events, such as personal interactions, using a piece of equipment, trying to build something etc. It is then up to the researcher to determine how such sources can be adapted, annotated or transcribed to ensure codability and comparability. As such, perhaps the best source for analysis, and certainly the source offering a path of low resistance, is a series of transcribed interviews. Interviews should be transcribed to the letter, as a single word can carry a lot of weight in the later stages of open and axial coding.

Further, the data that is to be analysed must offer some level of internal homogeneity and thereby offer comparability. That is not to say that different data points must agree with each other, but they must be in formats that offer similar properties so that the weighting of the extracted codes remain the same. For instance, if a researcher is to code a series of interviews together with some collected video, it is possible that one of these data sets will carry a very different set of codes than the other, making them more difficult to compare and code. A possible solution to this challenge is to perform the analysis of different kinds of sources separately, so as to create different sets of theory that will support, contradict or augment one another. However, all the data in a set should be

in the same form so as to offer comparability. In other words, not only should it technically be of the same media kind, for example video, but it should also be of events or situations that carry similarly codable properties. For example, it might be very rewarding to analyse video of how IKEA furniture is assembled in different countries, but it might not be desirable to compare video of driving patterns with videos of childbirth.

Table 2.1 Different sources and their applicability in grounded theory analysis.

Great Sources	Good Sources	Challenging Sources	Poor Sources
Interviews – transcribed!	Routine descriptions	Participatory/Passive Observation	Interview notes, abridged interviews
More interviews – go back!	Job descriptions	Video (except interviews)	
Academic texts	Internal documents, policy documents	Focus groups	
	Other unprepared texts, source code	Prepared statements, press releases, journalist work	

For this study, semi-structured interviews were conducted with a mix of purchasers and other users from different areas of the organization. The interviews were conducted in Norwegian, and the excerpts given herein have therefore been translated. The subjects and the selection thereof will be detailed in chapter 2.3. Most had extensive knowledge of the acquisition process, and where multi-year users of the ERP system, while one was in the process of being trained. Two users filled financial controller positions. Further, the instructor for a purchasing course in the ERP system was interviewed. The collected data was then analysed using grounded theory, seeking to uncover interesting relationships through the phases of open and axial coding. This technique implies that the structure of the coding is generated while the data is being read, followed by concept structuring by recurring themes (Orlikowski 1993). The conceptual framework developed was augmented by a document review to highlight some key differences between actual and planned practices, in a triangulation approach of sorts.

In the semi-structured interviews, the participants were initially asked to describe their own position and role within the organization¹. As a follow-up question, the subjects were then enquired as to what tools were used to support these activities. The question was not limited to the ERP system specifically or the organizations computer platform in general; rather, the users were allowed to discuss freely how they went about achieving their goals. Every subject eventually mentioned employing the ERP system, and at these points, the subjects would be asked about their experiences with the system, how it related to their job and in what ways they were satisfied or dissatisfied with the system. Especially, attention was paid to whether the system supported their desired mode of work and how the subjects might have developed their own methods of employing

¹ Interview guides are attached.

² All names in this thesis are fictional. Names of organizational units are anonymized or made up, but their

the system that did not reflect the planned use of the system. Further, the subjects were queried of how the interplay between communication with their colleagues affected or was affected by the order of work imposed by the ERP system.

As discovered, the experienced users had highly vocalized opinions on the ERP system and the ways in which it supported their primary work. It was not difficult to get colourful descriptions of the system from the subjects. Interestingly, the subjects never talked about the established workarounds or systems of augmentation (Gasser 1986) that were established without being asked about it. After enquiring, methods of working around or augmenting the system were discovered, and the subjects willingly shared their opinions as to why such procedures were necessary to support the primary work. Herein lies some of the strength offered by the semi-structured interview chosen for the data collection; the direction of every single interview could be changed to explore each individual use situation as much as possible while the interview guides simultaneously ensured that each interview covered roughly the same key points.

The interviews, which lasted approximately one and a half hour each, were thereafter transcribed in full. Coding and GT analysis was undertaken several times during the course of data collection and transcription, and in this way, data discovered in the earlier interviews shaped the latter interviews, an approach typical of grounded theory: *“In grounded theory, the analysis begins as soon as the first bit of data is collected. [...] Here, analysis is necessary from the start because it is used to direct the next interview and observations”* (Corbin and Strauss 1990, 6), otherwise known as *“theoretical sampling”* as put forth by Strauss (1987, 26-27, 38).

While beneficial to the data collection process, the effect of the continuously improving interviews also gives a slight preference bias towards the latter interviews, as they tend to contain more fruitful questions that were discovered as a result of earlier interviews and their analysis. This should become visible during the coding phases. It is by all means a desirable effect, but also one that the researcher must be mindful of – the volume of interesting data seems to grow through the course of the study, and is rather a sign of increased understanding on the part of the researcher than the ability to discover smarter interview subjects.

After a piece of data has been collected, the researcher moves on to the next phase of grounded theory application (Figure 2.1), which is the first of two coding phases, called open coding.

2.1.2 Coding – Open and Axial Coding, and Theory in Memos

The coding phases form the essential analysis parts of grounded theory. The coding aims to open up the data, and lets new concept and ideas come forth that are not explicitly stated in the data itself. In this section, we will take a look at how the phases of open and axial coding work together to

create suggestions of correlations, exemplified with data from this study. Lastly, we will have some discussion on the code memos, the prime product of the grounded theory process.

In short, during the coding phase, the researcher codes – or tags, marks, or categorizes – portions of the data. The way the researcher chooses to do this is highly individual, and while some researchers code word by word on a very fine level, others use broader scopes and code entire sentences, paragraphs or even chapters with certain codes. Similarly, what codes the researcher chooses to apply is also a matter of personal preference or belief. Some researchers stick to a low number of codes that are continuously re-used, while others come up with new codes that are slightly nuanced versions of earlier used codes. It is up to the researcher to decide what approach to take. In this study, coding was done at a fairly microscopic level, and a high number of codes were developed. The coding is very easily visualised by looking at an excerpt from the transcribed interviews shown in Figure 2.2.

The coding phases interplay with one another, and will feed back relevant information when the process is repeated. As we have already discussed, the continuous iterations and repetitions of the process are essential to grounded theory, as shown in Figure 2.1. These iterations happen on many levels of the process. Most noticeable in the figure is the major iterative cycle where the results from the coding, or analysis, will feed back to the data collection phases. However, the coding phases are also iterative and repetitive. While coding, the researcher will probably discover the need for new codes that might be refined versions of earlier used codes. To make use of the refinements offered by these codes, the researcher will have to go through the data again to apply the new codes. The process is repeated multiple times until there are no more codes coming forth – whereupon one says that the *codes are saturated*. As such, it is also important to not create any codes before the coding process is started, as this might restrain the researcher into coding with only these codes. The codes must come from the data itself, and not the other way around.

The authors of the literature on grounded theory are seemingly reluctant about giving away too many concrete details about the process, and Strauss goes as far as to suggest that it will be up to the individual researcher to discover his or her own ways of styling memos and codes. Because the grounded theory process is inherently linked to the thought processes of the individual researchers, the notes, memos and sketches need to reflect this individuality and become a product of the researcher performing the theory extraction.

Applying the grounded theory principles to the data, concepts were extracted from the interviews. Open coding, the initial stage of grounded theory, finds “*conceptually similar events/actions/interactions [that] are grouped together to form categories and subcategories*” (Corbin and Strauss 1990, 12). Further, the data was subjected to *axial coding*, where the different

categories are related to their subcategories while continuously being tested against the dataset (Corbin and Strauss 1990, 13). Lastly, through the process of *selective coding*, the codes are arranged around the discovered core categories and positioned as to further saturate the definition of these categories, lest they be rejected (Strauss 1987). These saturated categories then formed the basis of the structure of this thesis. Strauss suggests the continuous creation of integrative diagrams while coding, to assist with chiselling out the core categories and their relations (Strauss 1987, 170). One such integrative diagram is shown in Figure 2.6, and will be discussed later. We will now move through the coding processes.

As an example of the grounded theory process performed in this project, consider the following verbatim excerpt from the interview with the subject Eric, who is a teacher in the ERP system, when he was asked whether the development of homebrew spreadsheets is a feasible method of allowing the end user to get the view she wants of the data:

“From the standpoint of the user, of course; but, the challenge is: There are always suggestions coming in from users, subjectively, from the user: “I want it this way.” And then you have another user who wants exactly the same, but in another way, and then you have a third user that wants the same but in a third way. We have one common system and that’s why we have one way to do it.”

Applying the grounded theory process to this data, we can extract some openly coded categories and annotations that might form the basis of a theoretical memo – see Figure 2.2 and Table 2.2. With these open codes, axial coding is performed to produce the core categories that will be visible later in this thesis. The process is called open coding because it should not be restrained to codes that are already developed, but rather open to the continuous development of new codes and concepts.

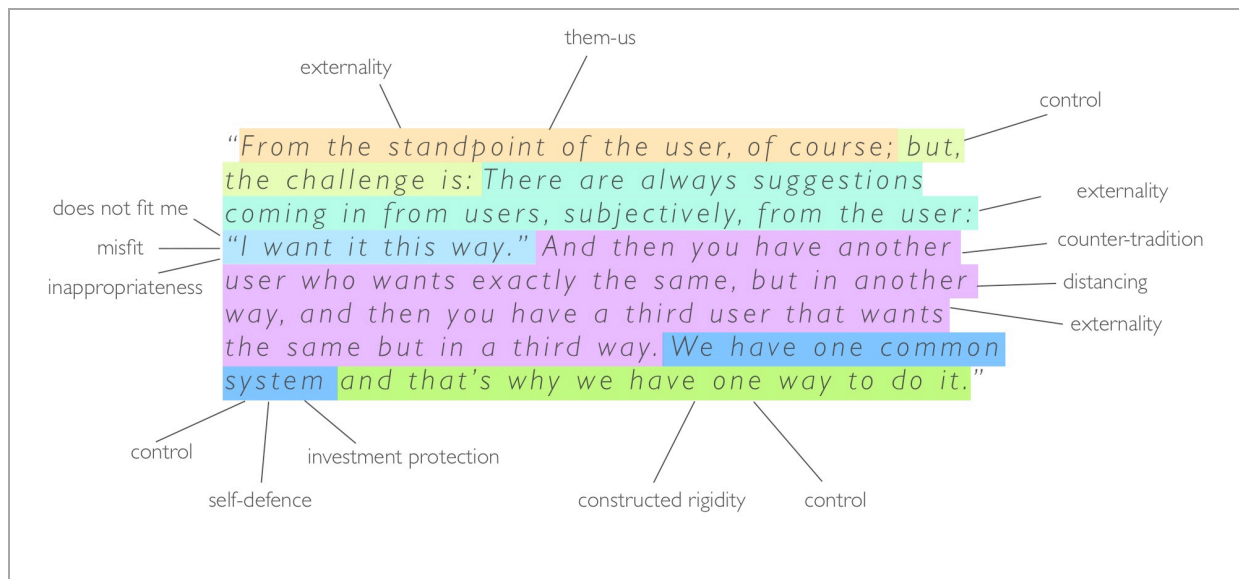


Figure 2.2 An example of codes applied to an interview excerpt. Same data as Table 2.2.

Figure 2.2 gives perhaps the most visual example of how open coding might look to the researcher. Some of the codes are re-used multiple times, while some are used only once. Some portions of the text are coded with multiple codes while others are marked with only one code. Other parts of the text, not shown, will not be coded at all. The same text is given in Table 2.2, where the codes are supplemented with annotations or notes that occurred to the researcher as the analysis was progressing. Such notes are, of course, highly internal and do not constitute any finished product – but they can be a meaningful way to start producing code memos.

Table 2.2 An example of theory extraction from an interview excerpt. Same data as Figure 2.2.

Data	Open coding / categories	Annotations
<i>From the standpoint of the user, of course;</i>	<ul style="list-style-type: none"> externality them-us 	Is there any feedback system? Why is there a distance between them and us?
<i>but, the challenge is:</i>	<ul style="list-style-type: none"> control 	Why is feedback difficult to handle? Why is it a challenge; should it not be part of the development cycle?
<i>There are always suggestions coming in from users, subjectively, from the user:</i>	<ul style="list-style-type: none"> externality 	The feedback of the users is 'subjective', it is not to be considered centrally?
<i>"I want it this way."</i>	<ul style="list-style-type: none"> does not fit me misfit inappropriateness 	The existing solution is not accepted.
<i>And then you have another user who wants exactly the same, but in another way, and then you have a third user that wants the same but in a third way.</i>	<ul style="list-style-type: none"> externality distancing counter-tradition 	
<i>We have one common system</i>	<ul style="list-style-type: none"> control self-defence investment-protection 	Are they too proud to admit rejection or defeat of the system? Why is the diversity of opinion neglected and thereby discouraged?
<i>and that's why we have one way to do it.</i>	<ul style="list-style-type: none"> constructed rigidity control 	

During the coding-process of grounded theory analysis, the perhaps most difficult choices lie in the development, application and saturation of different codes. Codes might appear in the middle of a long analysis, and they might suddenly seem very relevant to already coded material present in already-coded material. Or, codes might seem too rough or crude, necessitating the development of various nuances or distinctions. Then, what does one do with already-coded material? What about the material that does not fit into the newly apparent sub codes? The point, here, is that grounded theory-work is both iterative and nonlinear in that the process feeds itself new perspectives, and the progress is not measurable by simple means such as by looking at what page number one has reached during transcription. Further, saturation comes at different times for different texts or different topics, and the prediction of this point in the process is all but impossible. As such, one needs to realize that the grounded theory process has an implicit need for spontaneity, improvisation and pure, hard work with the data.

When the open coding process has reached a point of saturation, where repeated analysis of the data does not yield the production of new codes, it is time to begin grouping the codes during the axial coding phases. Axial coding can be performed manually or with the help of software. In essence, axial coding seeks to group the codes into code categories that contain a set of codes that have been found to carry some internal correlation or interdependence. This is where quantitative analysis can be of great help to the researcher. Axial coding should result in a set of core categories; groups of codes that carry internal resemblance or offer correlations, where the collection of codes are meaningful and interesting.

First and foremost, an overview of how often different kinds of codes appear in the data can be of great assistance. The researcher might find that some data sources carry a high number of a certain group of codes, suggesting that these codes might in some way be related. Or, the researcher might find that a certain group of codes are numerous in one of the data sources while absent from another, and vice-versa. This might imply that the presence of a certain effect negates another effect, suggesting correlation.

As mentioned, the data might also be analyzed using software that quickly applies a great number of calculations to big amounts of data, and then extracts and groups the results according to defined algorithms. Such analysis will be discussed in chapter 2.1.3.

When coding using grounded theory principles, it is extremely tantalizing to group codes and categories while still in the open coding process. This should to some extent be avoided because this grouping will taint or color the codes, and prevent them from being grouped in other ways later on. Simultaneously, not grouping codes will prevent early saturation of data – one ends up spending more time looking for relevant codes than doing actual coding since the codes themselves are

unstructured. It must therefore be up to the researcher to find a balance between sessions of open coding, analysis, and axial coding.

When axial coding is completed, the researcher might find that one has enough to begin producing code memos and theory. Should it be desirable, the researcher can also undertake a process of selective coding, where all the developed categories are gathered around a low number of core categories, that represent the main findings of the study at hand (Corbin and Strauss 1990). In this particular study, axial coding and selective coding was found to frequently happen simultaneously and not as discreet and explicit processes, which is why selective coding is not represented as a distinct step in Figure 2.1. This will surely be open to personal opinion.

What happens, then, to the theory that is hinted at or developed during all these various phases of coding? Code memos are a central part of grounded theory, and they represent the bridge between coding and analysis on one hand and theory and the written prose on the other. Code memos are short and fairly informal pieces of text that sum up the mindset, opinions and even feelings of the researcher at various times in the analysis process. Code memos can and should be written at any time during the coding. In fact, it is of immense importance to be able to break the open coding and axial coding sessions by producing code memos from time to time. This is difficult, since there might be sequences of "flow" in the open coding process that one does not want to interrupt or disturb unnecessarily. One must therefore remain attentive to the fact that the creation of code memos is the prime product of the grounded theory process, and should in most cases be given priority over coding when the desire to write a code memo appears.

The code memos can form the basis of a thesis under development, and in this study, they have played a major role in the development of the structure of the paper and the concepts presented. It is likely that more code memos will be produced than will be used in the thesis at hand, and the following rejection of a number of memos should not be avoided. Modern analysis software, as will be discussed shortly, luckily provide the researcher the ability to create links between these memos and their related codes and categories, offering a short distance between the repeated analysis of the data and production of memos.

During the processes of open coding and grouping through axial and selective coding, it will be necessary for the researcher to bear in mind that, as with any science, correlation does not imply causation, and that the methods will produce a number of false trails that must be disregarded although they can be strongly hinted at in the data.

Lastly, let us remark that coding in certain circumstances easily can be a pair activity. While coding was done alone for the production of this thesis, not seldom does one desire for a partner to assist in the analysis and interpretation of what is actually being said by the data. One can conceive

of the added benefit of multiple people providing their insights on the data at hand, either in real-time or by separately coding the same materials at different times. Luckily, software exists to assist with remote coding collaboration.

We have now seen how the phases of open, axial, and selective coding, eventually followed by the production of code memos and thereby hopefully theory, forms the crucial central elements of grounded theory analysis. Let us now take a look at how computer software can assist with this analysis, as was employed in this study, and afterwards round off with a discussion on how the final products can be integrated into working thesis material.

2.1.3 Programmatically Analysing and Visualizing the Data

Using software to perform grounded theory analysis greatly enhances scalability and flexibility, and encourages a high-volume output while coding, as opposed to a hand-written analysis where the never-ending development of new codes in the end can inhibit the process itself, and overview is easily lost.

All of the interviews collected for this thesis were coded and codes were structured using the research software application NVivo by QSR International. NVivo is designed for quantitative analysis of qualitative material in general, and thereby supports grounded theory analysis rather effortlessly. NVivo lets the researcher apply codes to segments of imported texts, whereupon the coded text either can be output grouped by the applied code – for example, by listing all sentences coded with the keyword “detective work” across all interviews – or the coded text can be used for computational analysis, as will be covered shortly. Multiple codes can be applied to the same text. Codes can be pre-defined, or created during the coding process – a crucial requirement for grounded theory research, since the researcher does not know what the data might reveal until the process has been completed.

The use of such an application becomes a great help in primarily the coding and later the analysis parts of the process. Coding styles vary greatly between individual researchers, and in this case, the introduction of a computer tool to assist with coding led to a plentiful increase in references coded, as can be seen in Table 2.3. The application also facilitates the programmatic analysis of the data in various ways, by either giving the researcher the option to query for word counts or various assigned variables, or by performing computational analysis on the data input.

Table 2.3 Number of codes assigned to some of the interviews, and the number of times these codes were applied during the text.

	Codes in use	References coded
Interview 7	69	434
Interview 8	89	369
Total		803

Table 2.4 *The 20 most frequently used codes along with the number of times they were used across interviews 7 and 8, sorted by sum descending, shortly after the first few rounds of open coding were completed.*

Assigned code	Interview 7	Interview 8	Sum
Too difficult	17	19	36
Avoiding Microwork	11	11	22
Constructed Rigidity	20	2	22
Control	19	3	22
Working Around	9	13	22
Telephone	11	10	21
Manual Routines	11	9	20
Not my job	11	9	20
Time-consuming	8	12	20
Competency	14	5	19
Manual Automatization	14	5	19
Competencial inadequacy	6	11	17
Fails to Automate	8	8	16
Future System	14	2	16
Detective work	2	13	15
Faith in the Construct	13	2	15
Backstage, No Knowledge of	11	3	14
Bad UI	9	5	14
Compliance	7	7	14
ERP System by Name	12	1	13

In Figure 2.3 we can see how NVivo has analyzed all the codes developed during the analysis of interviews 7 and 8, clustered together hierarchically by analyzing every single word assigned to each code and assigning the codes a Pearson correlation coefficient. This coefficient between two codes will be higher the more similar the words assigned to each coded are, meaning that codes that are applied to texts where similar topics are discussed should become grouped together. At the very least, sources where multiple codes have been applied to the same text will be tightly correlated because they contain the same words.

The codes with the highest internal similarity are in this way grouped together on a dendrogram, where the codes will be ordered pairwise by their similarity coefficient. Some 114 codes analyzed and presented simultaneously makes for an overwhelming visualization that does not lend itself to easy interpretation.

For curiosity's sake, we can illustrate the complexity of this calculation – and thereby the infeasibility of performing such a calculation by hand. The application computes the similarity between every two pair combinations possible. Given 803 references coded, as shown in Table 2.3, the number of possible combinations to compare equals the sum of every integer less than 803. Given that the sum of positive integers less than a given number can be calculated by

$$f(n) = (n - 1) * n/2$$

we get

$$f(803) = (803 - 1) * \frac{803}{2} = 322\,003$$

This thesis is not meant to serve as an exercise in arithmetic. However, the point is to show that the calculation proves to be of such magnitude that, first and foremost, it can only practically be performed by machine and secondly, the number of combinations tried is so big that the result surely and eventually will yield some result that might not be either explainable, justifiable or even reliable in a context like that of this thesis.

This brings us back to one of the core concepts of grounded theory – it is not the numbers themselves which is supposed to be significant, but rather, the representations or hints given by such numbers, analyzed by researchers with computers, and not by the computer alone. This is to say that the complete dataset does not lend itself to computational analysis or queries as such – it is up to the researchers to subjectively evaluate the codes, the first of which are given in Table 2.4, and then choose what one then wants to analyze further.

This means that for programmatic analysis to give a practical contribution, a more critical subselection of codes for analysis is needed. This is akin to the axial and selective coding phases discussed earlier. Let us take a look at two different examples of such selections. In Figure 2.4, ten codes have been selected fairly randomly for analysis. The resulting dendrogram is plainly structured with no obvious relations between the presented codes, meaning that the codes carry low internal correlation coefficients. The straightforward structure can be a sign that the selected codes bear less internal similarity or connectivity – or, that the selected codes are a perfect match and that the shape of the tree structure is a mere result of this. Considering the codes, however, the former seems the most likely. The result is therefore considered useless.

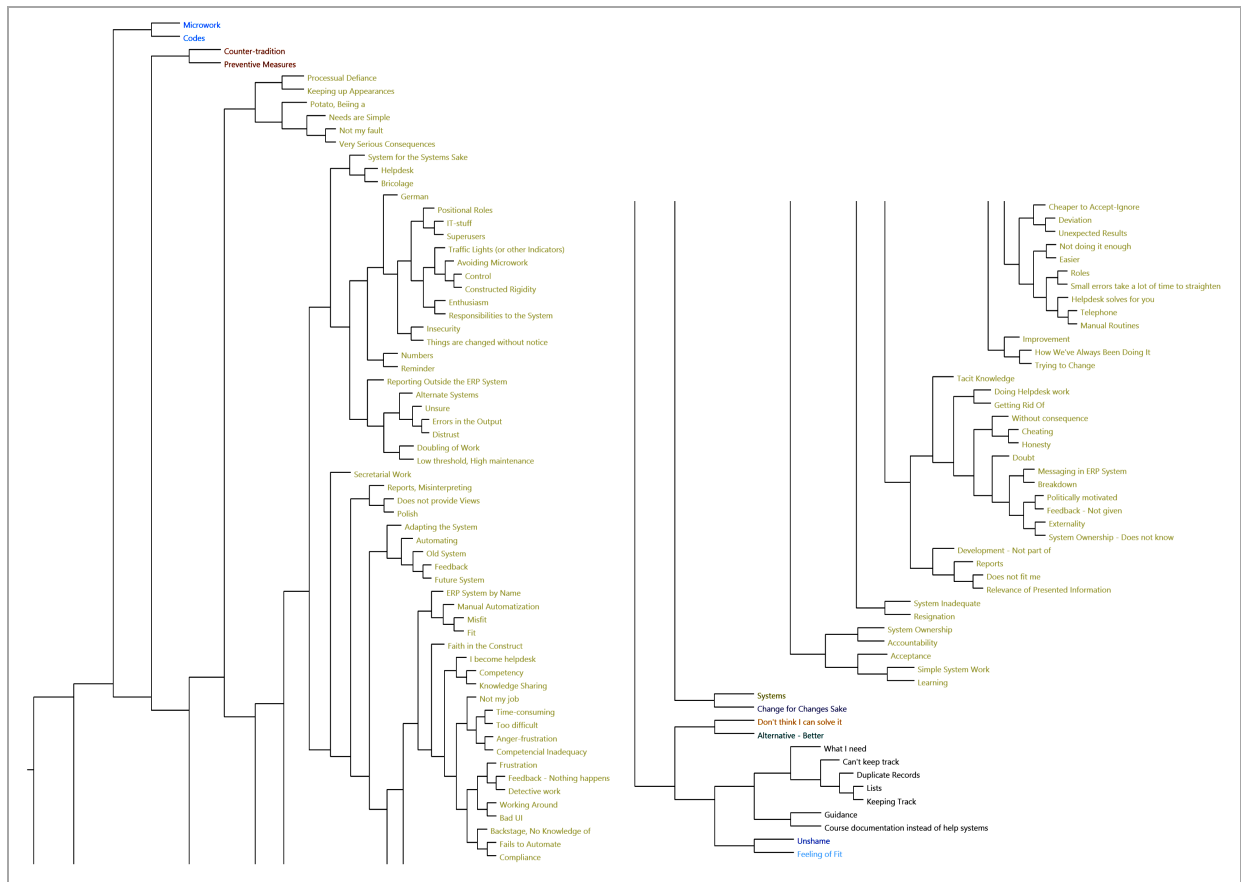


Figure 2.3 The codes developed after analyzing interviews 7 and 8, as clustered by the application NVivo. A subselection is sorely needed. The dendrogram has been split in two to facilitate the page layout.

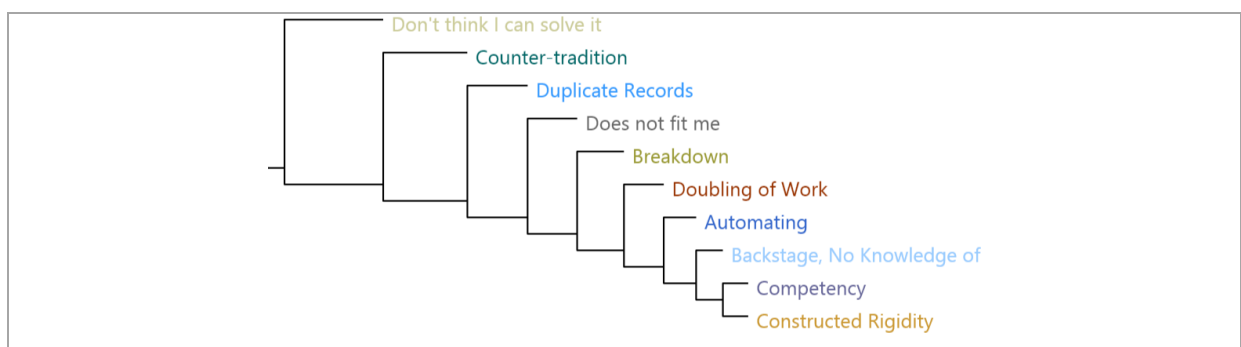


Figure 2.4 A random selection of codes for analysis yields a very straightforwardly structured dendrogram.

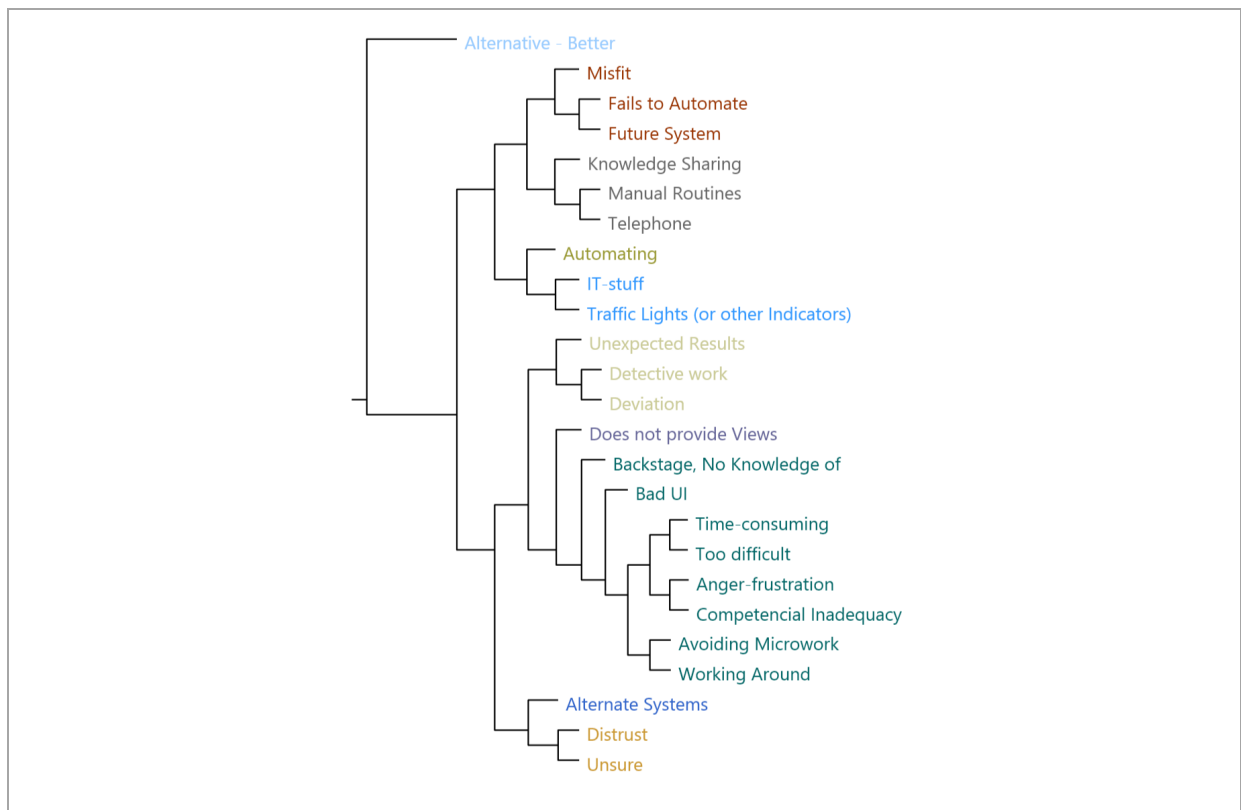


Figure 2.5 An example of a more critically selected subset of codes for analysis. A refined selection of codes for similarity analysis yields a much more interpretable and interesting result.

In this way we see that we cannot simply attack the collected and coded data with enough computational power to compute some interesting or interpretable results to write a thesis about. The researchers understanding of the findings and the ability to spot correlations and relationships remains as critical as ever. Computational analysis still remains relevant, however. Combining the interpretations of the codes produced so far with the insights gathered in the analysis work, another subset of codes can be analyzed for similarity – a selection of codes where the researcher carries an idea of their suitability for correlation analysis.

Given this more critical selection of codes, as is shown in Figure 2.5, we see a wholly different structure with clear groupings and hints at interrelated effects that were never explicitly stated during the interviews nor apparent to the researcher for consideration before the grounded theory process was applied. Suddenly, the combinations of codes are more than just a linear listing of the results – several strong candidates force multiple sub-groups to be created with stronger correlations. The application will structure the tree so that codes appear grouped by those who have the strongest internal correlations, and thereafter colored by which super-groups also have a high correlation coefficient.

Of course, for an outsider it is nigh impossible to understand the mindset of the researcher that gave names to these codes – code names are highly personal and contingent on a very high degree of subjectivity as they are formed in the mind of the researcher – so the raw data does not lend itself to easy analysis or scrutiny; however, such suggestions are indispensable to the researcher and provide much added value to the process.

It is crucial to note that programmatic analysis in no way exclusively formed the baseline for the findings of this thesis, nor was the application distrusted completely – as is prudent, the results provided by the computer was treated as suggestions for interesting relationships, while the final considerations still remained in the tradition of pure grounded theory work – through open coding, axial coding and the production of code memos.

We see that using software to assist with the grounded theory analysis is helpful, mostly due to the incredible complexity that quickly arises when a high number of codes are introduced, and further because the application supports quick extraction of data in ways that would be very time-consuming to do by hand or even in a plain word processor. Let us round off the introduction to grounded theory with some short comments on how to visualize the results using integrative diagrams, how to relate to the findings in the written text and finally some reflections on pros et contras on grounded theory in general.

2.1.4 Integration and Relation

One of the greater challenges of grounded theory lies in the integration of the discovered data with written text that forms the basis of the production of a thesis. As already discussed, the coding memos are the principal method by which the theory is integrated into a text. However, if memos prove difficult to produce during the open and axial coding sessions, integrative diagrams can be made that serve to illustrate the theory on their own.

Integrative diagrams are diagrams that reveal the relations between the different categories and the core categories (Strauss 1987). An integrative diagram is primarily a tool for the researcher, and serves as an addition to the code memos. One integrative diagram produced for this analysis is shown in Figure 2.6, included to give some insight into the culmination of the grounded theory process while also serving as a tool to explore the categories themselves. The uncovered categories have been placed in white boxes and these categories have again been grouped by their respective core categories, as explained earlier.

It was found that the two core categories *peripherality* and *control* both had what can only be interpreted as a causal relationship to the third core category *working around*. In other words, it seems that existence of the *working around* category was a direct or indirect result of the existence of the other two core categories. This can be considered interesting to note since it underscores the

causality of workarounds, suggesting that no workarounds are without reason or cause by themselves. The upper middle three categories remain ungrouped since they all bore no direct physically manifested symptom or effect, and did not relate to any of the core categories. The figure also includes the discovered aspect of constructed rigidity, encompassing rules and system rigidity that differs from the regular kind because they are not grounded in either law or common practice; they exist seemingly for their own sake.

The figure, then, shows how various factors contributing to the feeling of peripherality, called externality in the diagram, or how factors challenging the dimensions of control over the system leads to feelings of misfit, inappropriateness and insecurity. All of these states are then remedied by working around, seemingly solving the problem for the end user. Further, some peripherality factors lead to directly manipulative workarounds, labelled cheating, that again alleviate the end user of some form of burden or load. The resulting lower right two categories are outcomes of the workaround chain and are meant to illustrate how the workarounds seemingly solve problems for the users.

In this way, the integrative diagrams will help the researcher to visualize how the different concepts discovered might be connected. The integrative diagrams are not meant for direct inclusion in the resulting scientific text, unless they can be adapted to use as a visual aid while presenting the concepts. However, together with the coding memos, they form a good aid in the development of written theory and will provide good help to the researcher in the process. This leads to the next challenge of grounded theory, writing it up.

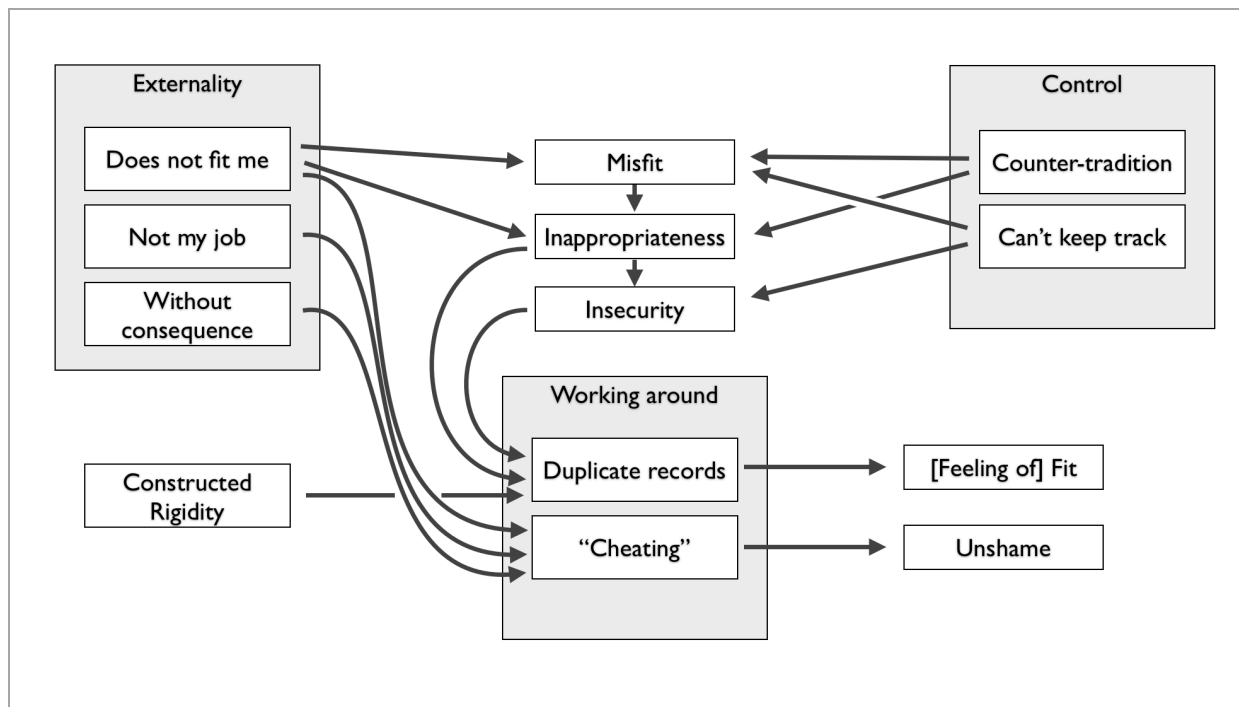


Figure 2.6 A proposed interpretation of the relationship between some of the observed effects: An integrative diagram highlighting the relationships between the core categories discovered in this analysis.

As mentioned, grounded theory does produce some prose through the production of memos, but the final presentation of the results and product remains up to the individual researcher. How does one present the data? We are lucky enough that many data sources suited for grounded theory analysis are excellent for inclusion in a text, as is the case with the current thesis – the use of transcribed interviews makes for easy inclusion, and lets the researcher select portions of their source data to back up their claims.

It is, however, not as easy to include all the codes and categories that might make sense to the researcher in the presentation, and it is therefore important that both the writer and the reader bear in mind that there is more data, evidence and analysis behind each excerpt than just the single passage of text chosen to represent each current phenomenon. Strauss (1987) in particular recommends the continuous interweaving of the analysis results given through codes and memos with selected pieces of source data, for example interview excerpts.

Through this introduction, we have seen how grounded theory analysis can be highly beneficial to an open-ended and interpretive study of a field of interest, summarized in Table 2.5. The methodology requires much more work than pure transcription of source data, but will as such provide results that are both grounded in the source data and also provide something more than what is explicitly said in the data itself. The methods are flexible and combine well with other

methods of analysis, should one wish to nuance the findings further. However, it is important to keep in mind that grounded theory will provide false trails, and the researcher must not expect to be able to use absolutely every single result the processes yield. Grounded theory requires the researcher to really get lost in the data, and while one cannot for sure know what one is looking for, one doesn't need to, either.

Table 2.5 A summary of highly subjective pros et contras for grounded theory methodology.

Pro	Con
(Usually) Great results	Takes a lot of time
Grounded Data	Takes a lot of work
The discovery of more than the sum of the data	Can give false trails
Free styled, suitable for many sorts of outputs	Can't use all the results
Combines well with other methods and methodologies	Needs immersion
You do not need to know for sure what you are looking for	You cannot know for sure what you are looking for

2.2 Organization and System

We have now taken a quick look at how the data collection and analysis has been undertaken in this study. In this part, we turn to an introduction to the organization that has been studied, its structure and operations, coupled with an explanation of the systems that form the crux of this thesis. After this introduction of the organization and its systems, in the next part, we will get an introduction to the subjects and how they were selected.

The organization under study is a Norwegian government entity totaling a five-digit number of employees. The organization is geographically distributed, with both smaller and bigger departments spread all over the country. Most of the departments are stationary and provide services in their own region, but others are mobile, relocating as needs change. The mobile units usually have a defined home base that they visit at set intervals for resupplying. Most of the workers in the organization work regular office hours, while some work according to various shift schedules.

Historically, the organization has been divided into several bigger divisions that each maintained their own structure, policies, accounting, administrative functions and IT-systems. As part of several initiatives for unification, effectivization and cost-reduction, many of the administrative functions across the old divisions have been gathered both organizationally and geographically into new departments that are shared across the entire organization – for example, most accounting functions have been removed from the divisions and gathered into a central accounting department, with the same happening for HR-services, supplies and IT. This restructuring process has been ongoing in its current form for at least ten years before the beginning of this study. All of the departments in the organization report to a superior department that holds central responsibility for the management of the entire organization, which in this thesis will be called *upper*

management. This upper management is still part of the same organization and must abide by the same rules and employ the same systems as the other departments.

As part of this initiative, some ten years before start of this thesis, the organization decided to centralize all of the various administrative IT systems that existed into one bigger Enterprise Resource Planning, or ERP, system that would become shared across the entire organization. The utilization of a shelf-ware ERP system, meaning a system that was developed by an external vendor prior to the acquisition, was deemed crucial as this would allow the organization to adapt a system that the vendor would continuously improve without risking to break any customization or adaption that otherwise would have taken place in the development. Further, the acquisition of an off-the-shelf-product would be a contributing factor to cost reduction and a speedier introduction. Lastly, the organization aimed at improving their central business processes by adapting such best practices that had been developed for the system. These notions are also mirrored by the literature (Ciborra 2000; Grabot, Mayère, and Bazet 2008).

It is in many cases interesting to study the introduction of a new system in an organization, but in this case, the introduction might become particularly interesting as the earlier divisions of the organizations employed their own systems that differed in size and function. As such, the introduction is not only an upgrade but also represents a unification across all the departments of the organization. This means that the introduction of the new system will not be felt the same way for all the workers employing the system, thus perhaps leading to differing outcomes of the introduction.

The new ERP system introduced in the organization has been split into two components; one represented by the classical user interface of the ERP system, and another more light-weight web-based user interface that would serve as a front-end for users that were not meant to perform complicated tasks in the system, such as employees registering what hours they have been working etc. Where distinction between the two is required in this thesis, the former will be called the ERP system while the lightweight web-based ERP-frontend will be called WEF. It is prudent to note that WEF does not store any data in itself; it is only a front-end to the ERP system.

The ERP system in question can be classified as a purely asynchronous system (Rodden and Blair 1991), as its design does not at any point require the simultaneous presence of any of its users nor does it offer support for such activities. Indeed, this thesis is limited to the study of asynchronous activities, save for short discussions of communication methods such as telephone or face-to-face meetings.

This introduction of the organization will serve as the backdrop for the discussion to come. Let us now take a look at the subjects that were interviewed for the study, before embarking on the presentation of findings and discussions thereof.

2.3 Subjects and Selection

The subjects selected for the interviews were gathered by making phone calls to various departments of the organization and requesting to talk with their purchaser or accounting controller. Response was generally divided; the departments that responded positively were keen on sharing their stories while those that did not wish to participate were fairly vocal about their desire not to do so. Further, four of the subjects interviewed were signed up by introducing the study at a purchasing course. This had the added benefit of ensuring a geographical and organizational spread of the subjects – the subjects are spread nationwide, and no two subjects from the study work in the same department or physical place.

The accounting department of the organization did not respond positively to request for participation in the study. No workers from what is described as upper management were invited to participate.

It is further important to note that all the subjects interviewed, save for one, were end users of the ERP system and did not hold any position of development or other sort of influence over the functionality of the system. While some studies might take top-down approaches to the integration of a system into an organization, this study positions itself as a bottom-up study, where the weight is given to the interpretation of situated practice and the opinion of the end users, which are considered experts in their fields, rather than analysing the systems intended functionality or vision. This is in line with the tradition of the Scandinavian approach to system design (Bratteteig 2004).

Adam² has been the manager of a warehouse of three people for five years. The warehouse is a clothing and consumable supplies warehouse for a unit of the organization. The unit that the warehouse serves is a mobile one, and is only infrequently present at the warehouse, resulting in a schedule of a very hectic two-day period while the unit is present and performing physical transactions, followed by a lapse of several weeks while the unit is not present. In this time, the warehouse receives requests from the mobile unit that the warehouse has to process internally, against other parts of the organization or against external goods suppliers.

Ben is responsible for supplies, stocking and inventory control aboard the mobile unit served by Adam's warehouse. He is in frequent contact with the warehouse to inform them of their needs and requests. Ben works on a cyclic shift schedule in which he has three weeks constantly on the job

² All names in this thesis are fictional. Names of organizational units are anonymized or made up, but their function and internal relations are represented appropriately to the extent it serves any purpose for the text.

while the unit is mobile and six weeks off while two other crews are on duty. Each shift is comprised of about sixty people.

Dina is also a purchaser in the organization, serving a remote site with logistics and procurement services. While only supposed to serve the one site, she is physically located away from the site, along with the logistics department that she belongs to organizationally. Dina is no longer employed, but has been hired back on a temporary contract after being let go because the department had trouble finding skilled applicants for the position. The temporary contract has been renewed four consecutive times, for six months each time. The site that Dina serves has only two regular employees present. While only formally intended to serve a single site, Dina serves a total of four sites because there are no other purchasers present to serve the other remote sites.

At a larger site of the organization, **Fabian** serves several hundred employees with supplies and inventory services along with seven others. Fabian's department also maintains a small warehouse where they stock the supplies. Only Fabian and two other colleagues perform purchasing. He has been in the position for over five years. Fabian tries to keep up to date with technological developments in the market and is a bit disappointed that the organization is so slow at adapting proven technologies like barcode scanners and wireless, handheld devices.

As a less experienced user of the ERP system, **Cindy** has recently been employed to work with development and public bidding of service contracts. She has a solid education in the theories of the field but has never used the ERP system before. She shares her tasks with about five others. She will never perform any goods purchasing, but will rather use the purchasing system to ensure payments for the contracts that will be established with vendors. While required to do so through her position, Cindy expects to spend very little time in the actual ERP system – most of the time will be spent working with the contracts.

Eric is the instructor in an internal training course in performing purchases through the ERP system, a job that he has held for several years. The course teaches both technical and regulatory aspects of the purchase process. Through years of hands-on with both the ERP system, the users and the developers, Eric provides some interesting insight into these relations that we will get back to later in the thesis. Interestingly, Eric got the job as an instructor by proving his skill at performing tasks in the ERP system while he was working as a purchaser. Eric provides insight coming from both the position of the instructor and that of the end user of the ERP system – a history that might have made him an instructor more skilled than had he come from a different background, perhaps a technical one.

Graham is a middle manager in a smaller department of the organization, having some six people report directly to him. His unit is one of five in the small department. Graham has been an

avid fan of the ERP system since its introduction, and wholeheartedly believes that the entire organization must be “*on the same system.*” He was a purchaser before being promoted to manager, and he has kept on the role as he is among the few in his small department who knows how to perform the purchasing in the system. However, after being promoted Graham has also been tasked with the approval of purchase orders for the entire department, which conflicts with his own role as purchaser – the same person cannot perform both actions, to prevent conflicts of interest. His unit has therefore been expanded with another person who will create and submit the purchase orders for Grahams approval. The new employee has not been trained in purchasing yet, and Graham therefore retains the task for the time being. The other four departments each have one purchaser who submits their purchase orders to Graham for approval.

Lastly, **Helen** performs the duties of a financial controller of another relatively small department of the organization. She has no managerial role, and describes her own primary tasks as administrative in nature – as she says, she is a bit of a “*potato,*” invoking the Norwegian saying which implies that one can be used for almost anything. Helen ended up with the role of purchaser for the department after taking the purchasing course some four years ago. Although her principal roles are controlling, auditing and reporting, she reports spending a lot of time performing purchase-related activities that seemingly take more time.

3. Findings and Discussion

Let us then turn to the findings of the interviews and a discussion of some of their possible implications. The findings presented here are the product of the grounded theory process, more generally based on some of the code memos developed during the analysis. The findings are indicative for the studied organization, and as they are presented they are also contrasted against relevant literature in an attempt to argue for their relevance and validity. Several, if not most, of the findings carry strong interrelations and these are explained where appropriate.

The results of the grounded theory analysis were linked with quotes from the interviews in order to shed more light on the presented phenomena than what a pure presentation of the grounded theory results would have done in itself. As such, the presented quotes must not be taken as the only proof of the findings, but should rather be viewed as a nuancing or exemplifying instrument.

In this thesis, the term “user” primarily refers to employees who maintain the role and perform the functions of purchaser in the ERP system. It is desirable to be able to generalize some of the findings so that they are valid outside of just the purchasing regime, and in these cases, the term will be extended to include all employees who perform work in the ERP system.

The presentation of the findings will begin with a short introduction to the general purchasing process, as described by the ERP systems documentation and then as explained by some of the users who are performing purchases. Then, some of the actual embodiments of the workarounds discovered will be presented, before we move on to the discussion on the characteristics of workarounds. After this, we will entertain a short discussion on technologies of accountability and ordering, and how the findings of the study can be related to this. The chapter ends with the unified presentation of the typology of workarounds and bricolages, and their characteristics, which will be based on the findings presented earlier in the chapter.

3.1 Process

As a frame of reference for the rest of the study, we begin with a short introduction to the purchasing process in the organization. The process will first be shown as it is described in workflow documentation for the ERP system, which will later be compared with findings from the interviews, contrasting the difference between planned system use and actual practice.

The process of goods and service acquisition in the organization is shown in Figure 3.1, taken from internal workflow description documents. In essence, the process requires the employees that are in need of performing a purchase to communicate their needs to a purchaser, who produces a

purchase order, or PO. The PO is sent to the local accountant who approves or rejects it, and the purchaser then sends the approved PO back to the employee who originally flagged the need, either as a PDF or by paper. This employee is responsible for sending the PO to the supplier. Upon receipt of the actual goods or services, the employee sends a confirmation to the purchaser that the order has been delivered. Meanwhile, the purchaser should have received a billing invoice for the order. When both the invoice and the delivery notice have been received, the purchaser links the two together and submits them to the accounting department for payment.

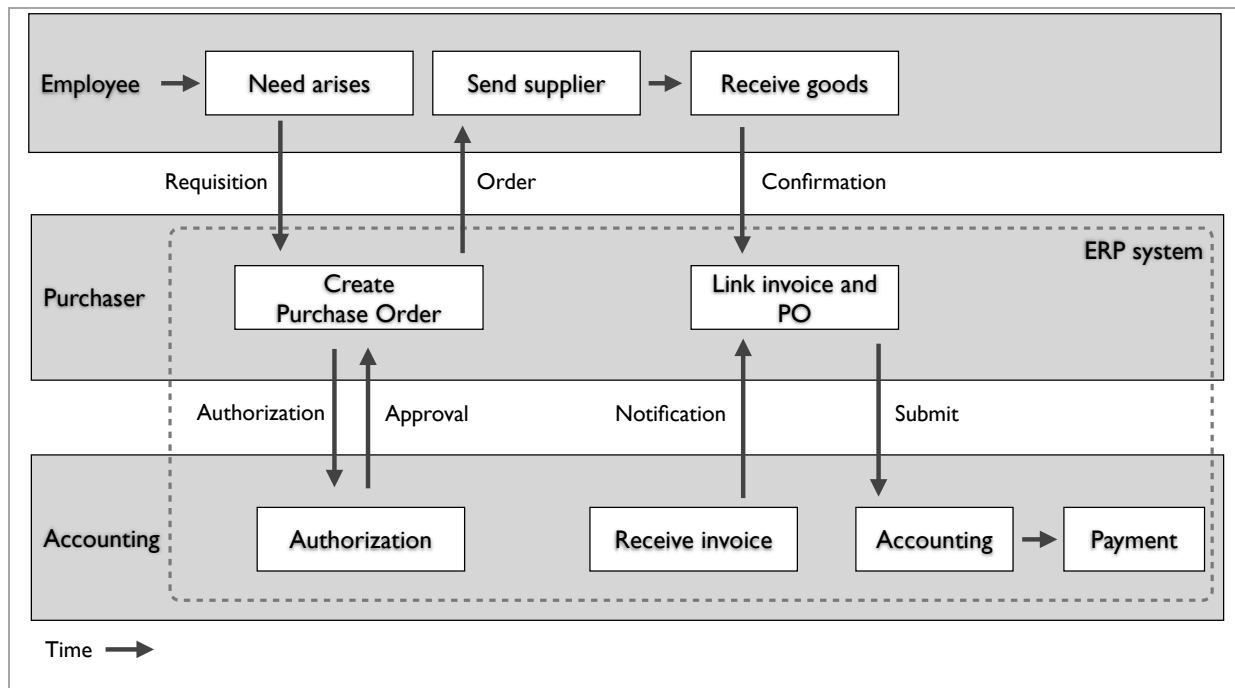


Figure 3.1 “Purchase to pay”, the acquisition process at the company, as described in internal workflow description documentation.

While appearing seemingly complete to the process designers, the process as described does not mandate any method of communication between the purchasers and the different actors in the process. The process description does not entail communication with other parties to the process: For example, the method of which the need is communicated to the purchaser by the end user is left up to the local organization to decide. We therefore seek to understand how the different pieces of information are conveyed across the different steps, and if at all the steps of the process are acted out as planned. Such knowledge will reveal details about the actual situated use of the system, giving insight into the difference between planned and implemented process, and real-world needs and practices.

In studying how actual situated use differs from planned use, we must acknowledge that this process model omits some elements that can be considered key to the process, such as how the

requests are communicated between the different actors in the process, how the purchaser would move data in and out of the ERP system and so on. As such, since the model is non-exhaustive, situated use will necessarily be subject to the differences given by the user interpreting and acting out the process. However, as we shall see, the differences brought in by the user might extend beyond just interpreting the model, as they are either amending or circumventing it.

Eric explains that, traditionally, the purchasing process used to be much more like that of a small company or even private organization: The company made a purchase, received the goods or services, and then received the bill some time later, whereupon the bill was registered in the accounting system, charged to the right budget and paid. While intuitive and relatable to private persons, this process did not give the company the proper insight or predictability of its own “*commitment plan*”, as Eric calls it. The process further allowed purchasers, or even end users, to perform acquisitions that were not approved by management or accountants. The seemingly more complicated process from Figure 3.1 was introduced together with the introduction of the new ERP system. Under the new scheme, a purchase cannot be made until a purchase order, or PO, has been made and approved by both management and accounting. At a bare minimum, the PO contains the name of the supplier, a description of the goods or services being purchased and the sum that the company has set aside for paying for the purchase. In theory, when a purchase order has been made, the arriving invoice will be automatically linked to the PO and paid in full if the PO has been approved and the goods has been marked as received by the end user. Eric refers to this as the “*main process*.”

Note that the mandated PO process plan must be seen as an idealized version of how the process should unfold, and it is reasonable to assume that it cannot be an accurate representation of the actual unfolding of events that is work for every single instance of the process – or as Schmidt puts it:

“For years, study after study have demonstrated, unambiguously and beyond any doubt, that the status of these formal organizational constructs in the actual course of work is problematic in that these constructs are abstract idealizations when taken as representations of actually unfolding activities.” (Schmidt 1999, 320)

However, it is not always feasible to produce a PO before the purchase: The necessary personnel might be absent, or the purchaser might not have immediate access to the systems required to produce the PO, or the nature of the priority of the purchase might make it impossible to follow established routine. It is therefore possible to perform a purchase using the “*post-hoc method*” where a purchase order would be produced for a given purchase after-the-fact that the invoice has already arrived. This kind of purchase is advised against, since it does not offer the budget

controllers the necessary insight into the commitment plan, and that one runs the risk of having an unapproved invoice show up. After the introduction of the new PO system, upper management has periodically set target goals for the use of main process POs, where it is required that a certain percentage of the POs are created this way. As the organization adjusts to the new procedure, the target is moved higher, starting at around 20 % and stabilizing at 60 % as of the time of this writing. Statistical reports are distributing detailing the various departments' compliance to the main process regime, creating a competitive environment across the organization.

The organization also employs a supplier relationship management intranet portal, or SRM, that allows purchasers to perform procurement from a catalogue. This portal is available through the ERP systems web-based front-end WEF. Some of the key suppliers have published their product catalogue to the SRM, from which purchasers can order items. The SRM will automatically forward orders electronically to the supplier, and integrates as a lightweight gateway to the ERP system. The SRM has not been regarded as a separate system in the context of this thesis, primarily because the purchasers interviewed rarely used the SRM, preferring to order directly from suppliers.

While the procedures established seemed reasonable and rational to the law-giving body, the study unsurprisingly discovered that the planned routine rarely could be complied with. Instead, changes in settings, requirements and approaches give instances of order processing that are composed of miniscule to more major differences from the established script (Suchman 2007). As such, the disparities between established procedures and end user behaviour, desire, setting and need necessitate the invention of actions that might become routine or are improvised on an ad-hoc basis. Such disparities are of particular interest to researchers because they represent the origin of concepts useful for further development of boundary objects (Star 2010).

We will continue the exploration of the embodiments of workarounds present in the study, before we discuss the characteristics and implications of these.

3.2 Embodiments of Workarounds

When analyzing a case with grounded theory, the results that are generated in the process can quickly seem abstracted from the material practices observed in the study. Let us therefore now take some time to illustrate and argue for the categorization of some of the actual embodiments of workarounds discovered. Some of these workarounds are fascinating in themselves, while others form indicatory backdrops for analysis later on. The workarounds presented in this section are but a few of the total amount of workarounds discovered in the study – which, of course, is also a matter of what one desires to classify as a workaround or not.

3.2.1 Communication

The subjects of the study readily started talking about the ways in which they interacted with their coworkers and suppliers, and opinions were generally in unison as to how these communications should occur. These findings provide an interesting reflection on the desire to have all incoming requests arrive electronically while being able to make all outgoing requests by phone.

Adam and Ben tell that they receive incoming requests by a variety of means, primarily e-mail, fax and phone but also face-to-face and informal handwritten notes. To a large extent, Adam prefers the written, asynchronous methods of communications, such as e-mail and fax, for incoming requests rather than methods that required direct interaction or response. This was contrasted with findings of preferred methods of supplier contact: In cases in which Ben for various reasons chose not to use the procurement portal, he preferred to contact suppliers by telephone to achieve a direct confirmation or feedback.

Adam commented that request for warehouse orders arrived on another network than the network that the ERP system is located on. This means that the operator has to punch the details of the purchase order into the ERP system running on another computer standing side-by-side.

As such, it is interesting to note that an extent of the communication relating to the process employed by the users of the ERP system take place outside of the ERP system itself. One can easily argue that the users function as a gateway to the ERP system for others who have no access to it; a valid point for those requiring purchasing services or otherwise needs the favours of the ERP empowered. However, it is remarkable that so much communication and coordination between users of the ERP system, even users who work on the same task or with shared task responsibilities, pool their communications or coordinate their activities in media external to the ERP system itself. Ideally, should not the ERP system support such coordination internally?

There are more examples of how communication can take place outside the ERP system. We recall that Adam is a warehouse manager providing a range of units with logistics services, and that Ben is the person responsible for supplies and inventory control aboard one of those mobile units. Neither Adam nor Ben knows that the other was interviewed for this thesis.

When Ben wants to order something from the warehouse, so that it will be ready to pick up the next time they come for supplies, Ben is supposed to set up an order in the ERP system that will then be sent to the warehouse for picking. However, Ben does not use the ERP system to submit these orders. Ben knows that the warehouse prefers receiving the orders in a more easy-to-use format, so he is using a Microsoft Word template that was given to him when he started the job. The template he uses contains a table where he inputs the items that he needs and their quantities, whereupon he e-mails the file to the warehouse while simultaneously archiving it himself. In this

way, he does the warehouse a favour by submitting the order in a more easy-to-use format, and Ben does not mind since either method is approximately the same amount of work for him.

Adam, the warehouse manager, is supposed to receive warehouse orders directly through the ERP system. Adam thinks that the ERP system is a bit cumbersome to use, but he knows how he generates a list of items for picking and prepares them for pickup. Orders seem to not come in through the ERP system, though. Most of the warehouse orders he receives come in by e-mail directly, usually as an attachment with some sort of table in it, detailing which items to pick and how many. Adam does not mind this, since he knows that the mobile units he serves are very busy, and he therefore grants them the leeway to go about their business as they see fit – after all, they are busy all the time while the conditions at the warehouse are much more relaxed. Therefore, he prints the orders from the mobile units, and then inputs them into the ERP system as a warehouse order, having the printed paper next to his screen. He can then retrieve that warehouse order back from the ERP system and have it generate a pick list for the items that the mobile units need. Adam needs to have the warehouse order in the ERP system because the orders will automatically update the inventory, which in turn will trigger alerts at pre-set levels so that more supplies can be ordered.

Ben, apparently, is not aware of the fact that Adam is inputting the data into the ERP system, which he anyway has access to himself. He believes that he is doing the warehouse a favour by supplying them with an easier-to-read-format. Adam, on the other hand, is under the impression that he is the one doing the favour, relieving Ben from the need to input the order into the system. Neither one has ever questioned the practice since both parties believe that it is for the better of the other party.

3.2.2 Cheating the System – Working Around the PO Types

When registering a purchase order, the user has to indicate in the ERP system what order type is being used: The user can either register a “standard order” or a “post-hoc order”. The standard order, used for *main process* purchases, will be sent to an accountant for approval before it can be linked to an incoming invoice. The post-hoc order can be linked to an invoice before approval, since the invoice has already arrived.

Since it is a stated goal to have as few post-hoc purchase orders as possible, the management regularly generates, distributes and publicizes reports detailing which departments have what percentage of post-hoc registrations. To avoid showing up on this list, Eric explains, the departments need to do as best they can to follow established procedures. Several of the departments found a way of working around this, however:

“Instead of using the post-hoc registration types they used the normal purchase order types.”

<And what were the results?>³

That it looked like it had been done properly.

<And how was the reaction to this?>

Well, really, it was positive, because then you'd done it right even though you'd done wrong. That way, when this was discovered, the report was changed to extract the data in another way so that the post-hoc registrations became visible after all."

In this way, the end user was feeding the system false information by indicating the wrong order type in order to keep the department on the positive side in the statistics. It is interesting to note that this actually implied an increase of work for the end user, since the end user then would have to wait for the purchase order to come back approved before it could be linked to the invoice – the cost of maintaining an image of process compliance. Looking at such a workaround is interesting, because it signals a misfit between the ERP system and the organization, and how the organization attempts to adjust to this misfit.

The workaround most surely conflicts with the formal system use, the idea being to use the order types to track process compliance, but the end users figured out how to use a modified version of the main process to their advantage and remain outside the statistics. As Eric explained, though, the behaviour was eventually noticed and the report was changed so that this workaround no longer gives any effect. The post-hoc order type still remains in the system, even though its point now seems moot, as it is no longer used for tracking compliance.

Another workaround was also discovered, where the end users would knowingly adjust the due-dates of incoming invoices so that a mass of expired invoices would not draw bad attention to the department. The organization also tracks and publishes statistics for expired invoices across the different departments.

"There were some cases [a few years ago] when every single department in the organization had expired invoices, save one department, that never had any expired invoices, not even months back. It becomes apparent that they themselves had gone in and changed the due-dates on the invoices. Because you can do that, and you should be able to do that. And the accounting department started checking, and if there was no explanation as to why the due-date was changed, it was reset to the original due-date. This is how those small cheating tricks are being tested. [...] It was sanctioned against."
(Eric)

³ Author's questions are given in <brackets>.

It is interesting to note that these two methods of working around are both present in areas that the departments having attributes measured and publicly compared in some sort of competition of process compliance.

Eric firmly believes that the statistics are of low value to the organization because they aren't tracking any real-life legal compliance; they are only tracking compliance with the rigidity of the system, a construct of the organization. In this sense, he claims that the implementation of a rigid system in some way absolves the end users of some desire to conform to the legislature: The rigidity of the system shifts the focus of the work task from legal compliance to system compliance. As such, if you comply with the rules of the system, it is assumed that everything is going according to the rules and that no other laws are being broken:

"[Look] at those statistics; I think those statistics are just nonsense, because they aren't telling you whether a purchase has been done right or wrong, it [just] tells you if you have been following the standard process – but you can create a purchase order for a million in advance, then it will be correct, without any competition or anything. But really, if you look at "correct" according to the legislature, then they are also doing it "correct" because they have a rigid system." (Eric)

Two practices must be upheld simultaneously: The organization acknowledges that post-hoc registration will always have to happen, and that the success ratio can never be higher than a certain percentage, according to Eric:

"The best reason I've heard for circumventing the main process was a dog trainer who had a dog that had gotten a metal spike through the foot: Then you go to the veterinarian. That's not even up for discussion."

We can see that even though all purchases can be post-hoc purchases, not all purchases must be according to the main process. In this way, users could choose to only keep to a single method for acquisition, but are forced to keep up with both models for acquisition and payment because of the synthetic requirement from upper management of 80% main process POs. In relation to having to deal with both the main process and the post-hoc registration, Dina exclaims that *"it would have been easier to work in the same picture all the time"* – referring to only having to learn a single interface for handling invoices.

3.2.3 Keeping Track and Restructuring

A common theme of the studies are the development of local methods for record keeping that are external to the ERP system and are handled outside the specified routines and practices. Recurring in such creations is the emphasis on what the user considers to be the primary function of their

work; the tool is not aligned with the desires of the ERP system, but rather with the actual objects that form the daily work of the end users.

The research has uncovered that such workarounds might be a result of several conditions: Firstly, lack of actual technical skill or know-how, as described by Star and Ruhleder (1994) as a “*First Level Issue*”, seems to have an effect where the end user will fall back to using traditional of safer techniques that the worker is more comfortable applying. Adam says that they “*don’t employ any extra lists or alternative systems [...] because we lack the fundamental computing skills*”

Ben and Fabian both expressed concerns that the ERP system does not offer sufficient overview of on-going purchases. In particular, almost every interviewed subject quickly and without question gave detailed descriptions of internally developed lists and other information spaces that was used to track the progress of different activities, even though activities were conducted in the ERP system.

Two different approaches to PO processing are shown in Figure 3.2 and Figure 3.3. Based on the descriptions provided by the interview subjects, the figures are only idealized versions of the situated actions, and surely not representable of the exact order of events for every single instance of the process.

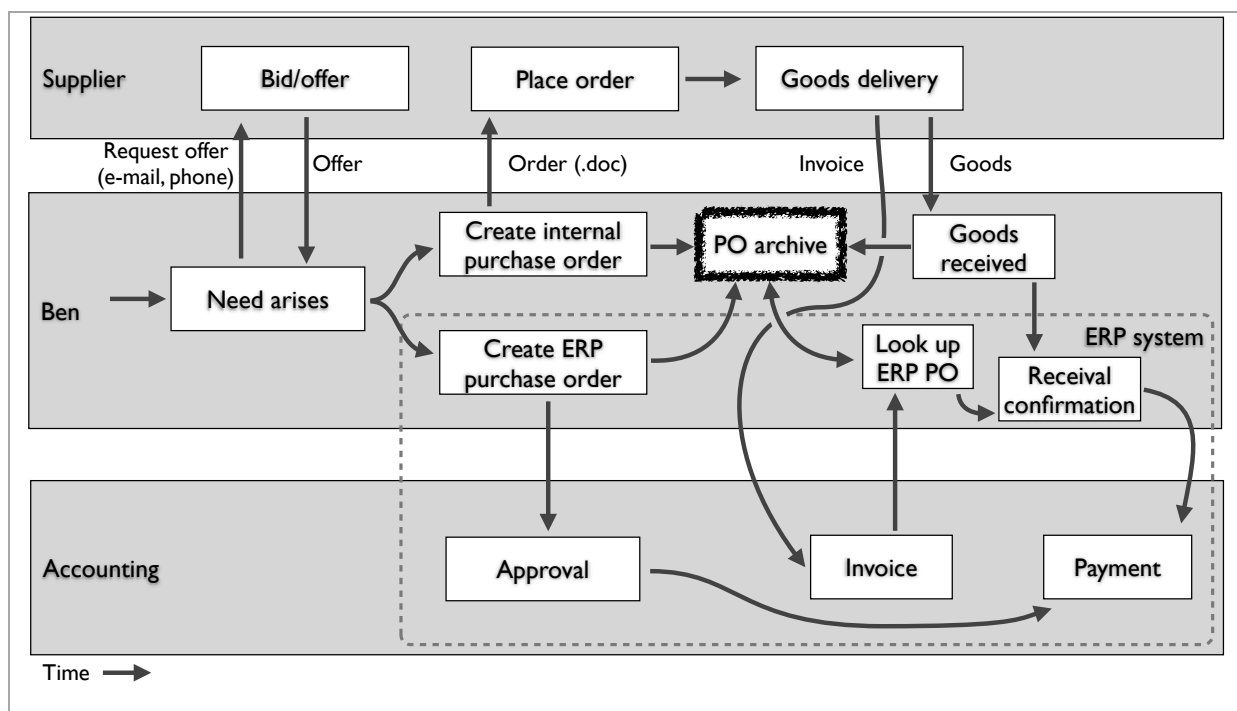


Figure 3.2 The purchasing process, following the main process, as performed by Ben. Note that the approval from the accounting department is never communicated back to Ben in any way, save for the eventual discovery that the invoice would never get paid.

Ben explained in-depth about not only internally developed lists but also internal purchase order forms that were used as a basis for entering the data into the ERP system. This form would carry an internally developed reference number that will be further detailed later. Ben is committed to the main process, and follows the established routines as best he can – with the help of his own order tracking system – and, while following the standard model, ends up spending most of his time outside the ERP system with his own proprietary ways of communicating with the suppliers.

Adam and Ben expressed belief in their own paper-based systems of control and insight. Adam commented that *“if I take it out on paper then I can sort it in binders by client and then look it up later”*. He later expressed that *“I wish I could print every picture in [the ERP system]. I can’t do that.”* and *“If I got a button in [the ERP system] called ‘Print’, God knows I’d use it”*.

The ERP system provides a list of the most recently created purchase orders, but this list is personal to each user and is not shared between co-workers in a department. As such, some of the external lists are used to facilitate the creation of a tool to facilitate the coordination and awareness creation among multiple workers, not unlike the common information spaces argued for by Schmidt and Bannon (1992).

Adam and Ben further expressed desires for notation systems to keep track of different purchases and other tasks. Ben and Fabian had developed notation systems that they used in their external lists, where each purchase was denoted with an identifier followed by a sequential number. For example, purchasing consumables in the Oslo region would produce the identifier OSLC003 etc. They were aware that the ERP system generated a reference number for each acquisition, but this number did not have any perceivable structure and was therefore considered useless.

The numbering schemes were observed in close relation to the external lists and forms – the lists would often contain both the internally developed number along with the reference number from the ERP system. Schmidt and Wagner (2004) found that such notation systems, developed and maintained manually, can be an important part of internal ordering work. The notations are created in a way that is meaningful to the user and not to the system – the development of such notations can thereby be seen as a response to the lack of such meaningfulness in the notation from the system.

Ben uses this notation scheme to track purchases in their own lists outside of the ERP system. These lists are shared across three different shifts crews on his mobile units, so that the other two workers can take over his job when he leaves and follow up the purchases. He needs to maintain this separate archive of purchases because the ERP system does not facilitate sharing data structured in this way between multiple users.

In addition to his new and internal notation system, Fabian's department has purchased barcode scanners and label printers to keep better track of their inventory. This equipment has not been integrated into the ERP system, because as far as Fabian knows, the system does not support it. Fabian and his colleagues have, however, experimented with the scanners and printers in combination with the notation system and their own tracking lists. They have not yet been able to figure out in which ways they can use these tools, but it is clear to Fabian that once they do, they will be able to achieve a much higher level of efficiency. The equipment remains unused.

Dina presents a less complex view of how she keeps track in the ERP system. She tells of how she prints two copies of every invoice she receives, so that she can file them both by supplier name and in a chronological order. This view is not present in the ERP system, according to her. It is in fact possible to extract the desired data in both of these ways. Dina's solution to not knowing how to view the data in this way, then, is to make a paper-based shadow system. While this system might seem simplistic, it is interesting to note that it is among the few ways of working around that are actually independent of the formal system in all ways – no matter how the formal system should be changed or updated, Dina's duplicate records will survive and can be kept up. This is not necessarily the case for other instances of workarounds that depend on references to objects in the formal system, for example by purchase order number and so forth.

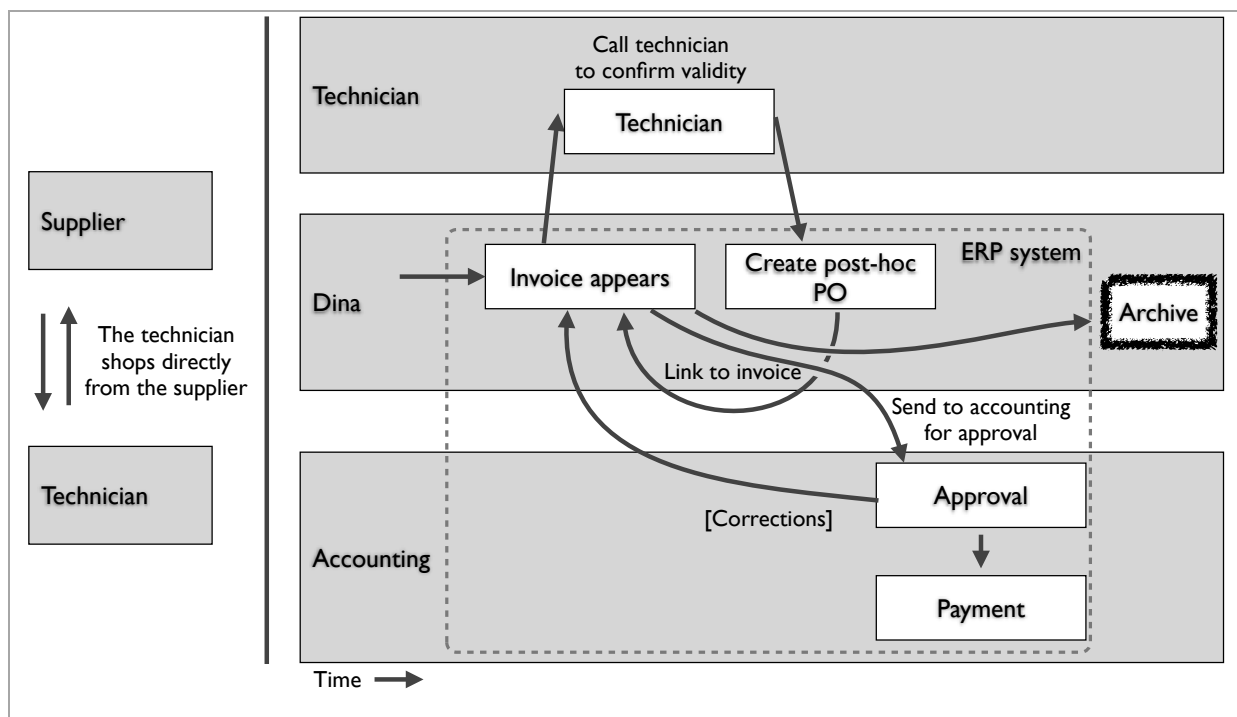


Figure 3.3 The purchasing process, post-hoc, as performed by Dina. The technician seemingly has a carte blanche to perform purchases as he sees fit without notifying Dina.

Dina does all her purchase processing post-hoc, against the will of upper management, who requires a certain percentage of POs to be according to the main process. She has little or no choice in this, because the technicians that generate most of her purchasers never contact her before the purchase is made, resulting in invoices appearing out of thin air. She has to call the technician for every single invoice that appears and confirm that the invoice actually relates to a purchase that the technician has performed. However, she spends almost all her working time in the ERP system, save for the minutes it takes to file paper prints of all her incoming invoices in two sets of binders – one chronologically and one grouped by supplier. Schmidt and Wagner (2004) explored ordering systems, also coming across binders, and remarked that while the binders do provide a system to order the pieces of data that can be categorized, there will always be some things that remain outside the classification in use. To Dina, then, this ordering system must be perfect, since she will never experience any piece of data that does not fit into either of these binders.

The image shows a hand-drawn spreadsheet on lined paper. At the top, there is a header row with the following columns: Dato, mndref, LOUR, LEV., AKTIVITET, KORT BESKRIVELSE, ANT, PRIS, RENU, INNHJ., BEHANDLET, and PROVANT/FORBUD. Below the header, there are several rows of handwritten data. The first row under the header has 'Høstet' in the 'Dato' column, '10' in 'mndref', 'Renne-10' in 'LOUR', and 'Løst overfor' in 'LEV.'. To the right of this row, there is a note 'Type order' with an arrow pointing to the 'AKTIVITET' column. Below the first row, there is another row with 'VEDLIGTHOLD' in 'Dato', 'DEUSTOF' in 'mndref', and a checkmark in 'LOUR'. To the right of this row, there is a note 'FAKT. BLAGM.' with an arrow pointing to the 'BEHANDLET' column. The table is drawn with vertical lines for columns and horizontal lines for rows.

Dato	mndref	LOUR	LEV.	AKTIVITET	KORT BESKRIVELSE	ANT	PRIS	RENU	INNHJ.	BEHANDLET	PROVANT/FORBUD
Høstet	10	Renne-10	Løst overfor								
VEDLIGTHOLD	DEUSTOF										

Figure 3.4 Hand drawing of Fabian's internally developed spreadsheet to keep track of purchases and their related activities. The spreadsheet was praised because it offered flexibility while imposing few restrictions on data input. The spreadsheet does not process the data in any way, it only structures and stores the data.

Several of the subjects explained in detail about the internally developed systems for record keeping and follow-up. In itself it is of course interesting to observe the different ways in which the departments have decided to fashion their own tools for these purposes, but it is also interesting to observe the way in which the users talk about these tools. For example, when asked about the list in which the department keeps their duplicate shadow records of their purchase transactions, Fabian explains with great joy and even pride. He pulls up a blank sheet of paper and draws the list with all its features, and elaborates on how this tool has become essential to their organizational model.

The drawing is shown in Figure 3.4. In addition to the insights gained by looking at the drawing itself, it is also interesting to note the enthusiasm, conviction, pride and belief that was conveyed while he was drawing. Remember that Fabian has been using the ERP system full time for five years. In conclusion, he describes his vision:

"[the ERP system] should be more like this." (He points at his drawing.) "There is an incomprehensible amount of buttons in [the ERP system] that you never learn. On a course I will learn one or two new buttons; what about the other 100?"

Looking at his statement, we see at first pride and faith in the internally developed system, suggesting it as an improvement to or even replacement of the ERP system. But then, the pride turns to insecurity and distrust; it seems that the situation is not only that the internal system is better, but also that he does not know how to achieve his desired outcome in the ERP system.

Looking at the contents of the drawing itself, though, we observe interesting properties that deviate from the structure of the purchase orders in the ERP system. For example, the ERP system treats every order line of the purchase order the same – they are all abstracted objects that share exactly the same properties, no matter what physical object or service they may be intended to support. To Fabian, though, a purchase is less about the act of purchasing and the properties around it, but obviously more about what is actually being purchased: His list contains columns such as "provisions", "fuel", "maintenance" and so forth, in addition to the "supplier" and "price" columns also found in the ERP system. He does not see relational tables, indexes or unit numbers, then, he sees the core objects and services of the primary work that his department serves, and because those are the objects that are important to him, it will be those objects that are quantified in his own construct that he has put together with what he had at hand – his own bricolage.

External or shadow lists are not unknown in the context of workarounds. One cannot overstate the importance of understanding the orientation of such workarounds, as we have seen in this section. The creations tend to be directed towards what the users consider to be their primary duty of work, an excellent example being Fabian's spreadsheet – it emphasises the real-world content of the objects it abstractly represents, in stark opposition with the ERP system, which desires only to represent the values important to it – cost, quantity, status etc.

Schmidt and Wagner (2004) discuss how users create, maintain and distribute their own systems of structuring and ordering information, and that these systems can grow and be adapted in a project, while still being re-used in future projects. These properties of orderings systems offer great economical benefit in that the development of context and structure can be re-exploited in future implementations. In this case, it is clear that the development of such systems can greatly benefit the end users in their work, and that the creation of these structures do live on beyond the

individual purchase orders and even as employees come and go. However, a lot of development of such systems must take place concurrently in the entire organization, and the question remains if all the work that goes into the development of these systems in some way can be shared and exploited across the organization.

It is worthwhile to note the ease with which the users jump to the conclusion that they have to create such systems; if they sense a shortcoming of the system, they augment the system with their own workarounds and bricolages of utilities and knowledge. But while all of these lists and records that are created are rather passive and must be kept manually, there exists another kind of workarounds where the end users bring the data out of the ERP system and into their own creations: They create their own bridges between their own tools and the formal data of the ERP system.

3.2.4 Bridging to the Formal

We have seen how users create libraries and lists of their own when they are in need of representing objects that the ERP system does not represent in a desired fashion, especially objects that are close to the primary work of the end user. Common to all of these creations is that while they are clever and surely useful to their creators, they require manual record-keeping and updating to keep up with their purpose.

Workarounds and different forms of bricolage have obviously not gone unnoticed by the developers of the ERP system, who have provided an add-on to the ERP system that lets the end users export data from reports in the ERP system into other applications, such as a spreadsheet. Using this tool, the user can extract data automatically, repeatedly over consecutive time periods, for example to make updated monthly reports.

Specific fields from different reports can be inserted into fields of a custom-built spreadsheet-application, whereupon the user can employ many common features, such as calculations, pivoting, graphing etc. The tool, however, is fragile, and is dependent on the spreadsheets being created in a predictable way. If the end user moves the data around too much, the tool will no longer be able to figure out where the data is supposed to be inserted, and will fail. The user is thereby both empowered and restricted by this tool – he can use the tool to automate common procedures and repetitive tasks, and perform calculations and computations that are unavailable in the ERP system in itself, but is simultaneously restricted from applying the entire breadth of his skillset, since the tool will break when it can no longer understand where the data is supposed to go or what is being asked for. Therefore, the tool comes across with a high perceived cost of maintenance, requiring the end user to tinker around with various options until the tool will deliver a result that is satisfactory.

Knowledge of this tool is far from universal in the organization. About half of the subjects interviewed reported knowing of this tool, and only Helen reported that she was using the tool actively. This contrasts with the fact that all subjects mentioned employing some form of workaround, suggesting that while the desire for the creation of tools outside of the formal system is strong, the wish to have such creations integrate with the ERP system is low – perhaps simply because of lack of knowledge of the tool, lack of skill in implementing the tool, or perhaps as a conscious choice in order to have greater control over the data that is being input.

The existence of this functionality is interesting in itself. Firstly, the ERP system seeks to be universal in that it in itself asserts to provide all the reports and features necessary to conduct the tasks of all the roles that are in existence in the system. If the ERP system is functionally complete, then from where does the need for the data-extraction add-on come from? When only such a small amount of the users reported using the tool, it cannot be claimed to be an essential part of the workings of the ERP system for all users. The purpose of the tool then comes under question; is it perhaps the vendors attempt at facilitating the creation of tools on the part of the end user – are they simply inviting the bricolage? The users have no knowledge of the tool that comes from formal training, only from experimenting, which makes the product of this tool just as much of a bricolage as some of the other creations studied in this thesis.

The tool, then, provides an invitation from the vendor of the ERP system to work around the perceived shortcomings of the system through tinkering and the creation of bricolage. Interestingly enough, the majority of creations that store and structure data from the ERP system does not use this tool, even though about half of the users seem to be aware of it. It could be because the users are simply unaware of how to employ the system, or it can be a conscious choice, electing to have the construct standing on its own legs, disconnected from the formal system. Regardless, the only open invitation from the ERP system to bricolage remains largely unused.

3.2.5 The Good Intentions of Disservice and Disregarded Feedback

While the organization provides a technical help desk department that is available both by phone and through the intranet, the organizations central accounting department - the department that receives, scans, and processes invoices - is also available by phone to support the end user. As such, the responsibility to provide assistance to the end user in case of problems is somewhat split across a technical help desk and the accounting departments help services.

The principal function or purpose of the accounting departments help service organizations is questioned through the results gathered in the interviews. For instance, a user who was routinely experiencing the same kinds of problems with the payment of purchase orders in the ERP system found that the accounting organization usually preferred to override the process and correct the

problems with the purchase order themselves, rather than showing and teaching him how to either solve the problem or avoid the problem altogether in the future – as Helen explains:

"[When processing purchase orders] you always have to go in and process it manually, and when you are processing it, if you click the wrong [button] once, then it will no longer manage to connect it [to the invoice], and then you're stuck there with the invoice - just that invoice - and even if you type the purchase order number there, the number, then it will not be able to tie it to what you've already made, and that I've never learned how to solve, and I must call [accounting] every time that happens. So a small error there makes you unable to solve it.

<When the error occurs, does [accounting] solve it, or is it something they will show you how to solve, or how...?>

No, no, they say 'We'll take this', and then it disappears.

<And then they solve it for you?>

Yes. Yes, yes.

<So you don't learn how it should...>

No, it's almost gotten to the state where I don't think I can do it, that when I get into this state it will lock up and that I with my access cannot do this."

By not teaching the end user how to either solve the problem themselves or avoid it altogether in future operations, the accounting organization not only upholds the competencial inadequacy at the end users position, but rather teaches the end user, implicitly, that he or she is in fact unable to solve this problem and that this is something that someone with higher access permissions has to solve.

Using Bateson's levels of learning, Star and Ruhleder (1994) characterize three levels of infrastructural complexity, in which the recovery from a higher-level situation costs more time and effort than from a lower-level problem. In this situation, the help desk actually transform the problem into a higher-level one, creating a *double bind* when the system assumes that the users should do it while the help desk repeatedly does it for them.

As such, the end user will not question the fact that the accounting organization chooses to solve the problem on the end users behalf - it has been made clear by them that they will have to solve it for them - and the end user might then seek alternate ways to avoid the problem again in the future, thus spurring some attempt at a workaround or other adjustment.

Another interesting perspective on this issue can be attained by interpreting this situation as an extension of the relationship between the end users and his or her own challenges. The end user is generally reluctant to engage in activities that are more laborious than their alternatives, for example as shown through the distinction between the use of purchase orders according to the main process versus the post-hoc registration of purchase orders. Consider this choice as a choice of the path of least resistance. In this sense, the choice faced by the help desk organization between solving the immediate and at-hand problem, as opposed to instructing the end user on how to do it properly and hoping to avoid the problem in the future, is in its essence the same form of problem only on different levels or at different stages. Helen said:

“Instead of helping they know... they’re sitting... by the time you’ve explained the problem to [the accounting organization’s helpdesk] they’ve got all the information up on their screen already. And then they can push the necessary buttons and it’s gone from this world. I think maybe they’ve given up, maybe, and [they’ll] just solve it to get out of their world.”

Taken further, it can even be envisioned that this behavior might actually be desirable for the help desk organization. Should the help desk operators be measured and incentivized through the reduction of the length of incoming calls - shorter calls and faster fixes are better - then the help desk organization would not desire to have more skilled end users that would solve the most difficult problems on their own and only present the more challenging, and thereby maybe time-consuming problems, to the help desk. Similarly, instructing the end users might lead to a lower volume of calls that can have a negative impact on the organizational reviews of the help desk organizations.

This is not entirely unlike the findings reported by Orlikowski (1995), where the performance requirements and attributes that the help desk operators are measured by, give the operators the desire to respond to a higher volume of calls of shorter duration, as opposed to spending time teaching the end users the correct practice, which might reduce the long-term volume of calls. This thesis is not the place for a lengthy discussion of such issues, but it remains prudent to be aware that different parties can be motivated by the different perspectives on the same problems, or at least remaining somewhat disinterested in the obliteration of these situations.

Regardless, the way the helpdesk organization chooses to solve the problem never seems to benefit the end user in any long-term sense. This creates a deep-seated feeling with the purchasers: They cannot work around such problems themselves, but have to enlist the assistance of some other party in order to get the problem solved. The way the helpdesk acts thus turns what would otherwise be a small sidestep of the routine on the part of the purchaser into a workaround.

All of the surveyed users were in some way of the opinion that the ERP system had one or more shortcomings that could be easily remedied by developers if they would take the time to listen to and consider the ideas that came from the users. Especially, some users were of the opinion that some reports in the ERP system provided blatantly false information, and that they thereby could not use them. Users generally felt top managements pressure to use the ERP system for what it's worth, but were unable to comply with this because they valued the correctness of information more than their desire to please top management by using the system, or as Helen said:

"Yes, that's what... that's what they... from the top there's a lot of, 'don't use Excel by itself, but use [the ERP system] for what it's made for'... that's not very easy when the data you put in don't come back out in the correct way. So... I use [the ERP system], but often when I hand out reports to the branch managers here, then I have to use Excel a lot, because of... if I go down to each branch's accounts and consumption and what I have left on the result, then it's not correct."

Here, Helen uses the ERP system to process the data as she is asked to, but in her opinion, the data that is returned from the system is simply incorrect and not representative of her situation. She then opts to continue feeding the same data into the system, staying true to the wishes from top management, but disregards the output from the system and instead generates her own report for internal distribution. Central to several of these stories is the fact that all the users claim to have, at some point, notified somebody else – be it the help desk organization, or the accounting department – but no improvement has been observed.

It might be the case that the user has lodged some form of complaint with the developers, but it went ignored and no improvement took place. However, there are three plain reasons why the existence of this phenomenon might still be the blame of the end user. Firstly, the user might be under the impression that some notice has been given, while in fact nobody has been made aware of the situation. Second, the user might have mentioned the problem in an off-hand fashion or in way that did not make sure that the help desk representative realized this was an actual complaint or improvement suggestion, thus leading to a situation of shared misunderstanding where the user believes this has been reported while nobody has made any note of it. Third, the user might simply be working the system in the wrong way, and the representative from the help desk has dismissed the report as a user error.

To us observers, these different causes are fairly distinct and carry different consequences. It is, however, thoroughly irrelevant to the end user whether she is the actual cause of the problem and if she has contributed to its solution or not: Since the user is of the belief that it is not her fault and that she already has put some effort towards solving the problem, she will not take any more

steps to try to remedy the situation himself, and might not attempt to seek any more remedies since she believes that the developers are disinterested in receiving any feedback. Additionally, this might deter the users from providing any further feedback, since they are not of the opinion that feedback is heeded, regarded, or perhaps even read in the organization. This also contributes to an alienation of the end users, since nobody seems to listen to them – even if they actually never spoke, or didn't speak to the right person – they end up perceiving themselves as outside the target audience of the system.

Workarounds that stem from the belief that there is something wrong with the system thereby come about; the workaround becomes a crisis solution to a permanent problem. It is perceived that nobody listens to feedback – or users don't know how to provide it, problems are not corrected, the production of erroneous data continues and workarounds are created.

3.3 Characteristics of Workarounds and Bricolages

We have now seen some of the embodiments of workarounds together with some of the conditions under which they come about. We will now take a look at some of the characteristics that are common to most, if not all, of the workarounds found.

Characterizing workarounds is inherently challenging because the definition of the workaround is inherently challenging. More than being just a set of characteristics of workarounds, this chapter will also characterize some of the conditions under which workarounds and bricolages come about and flourish. Firstly, perhaps the foremost attribute of a workaround is that it represents a situation in which computing simply does not fit.

3.3.1 Misfit

It is not hard to imagine that the primary motivator for working around a computer system might be situations where the computer system simply does not offer what is required by the end users. This is the principle presented in Gasser's paper on workarounds: Misfit occurs when there is "*slip or slack [in] the computing resource which is supporting some primary work*" (Gasser 1986, 212). Slip and slack refers to the respective under- or oversupply of computing resources, or general misalignment – there is too little, too much or simply not the correct kind of computing resources available. We will take a look at how this definition might be too pessimistic, and how misfit can be a desirable effect that can be an incentive for creation and bricolage, leading to an improved outcome.

Gasser's definition of misfit lends itself to being interpreted as a defined through negation – as if saying, "*misfit is everything but good fit*". This, then, would imply that misfit always carries a negative undertone; suggesting that it is undesirable or that one should seek to eliminate it. In the light of this study, this definition of misfit becomes too dystopian.

It is true that the workaround senses an opportunity for existence when there is a misfit between system and the user/organization, and the workaround will come into to life as long as the system does not provide a method of fitting or because the user does not have the necessary skill required to fit the system. Such fitting work does not only encompass changing attributes to make the system more usable, but also the desire or knowledge of how to acquire skill in the system.

Discussing the nuances of workarounds and fitting work, it is important to realize that not all adaption and fitting is working around – an ERP system necessitates local adaption and fitting work, both on the part of the organization and the system. This is unavoidable; the nature of individuality makes for no single system that can achieve a perfect fit with no adjustments or tuning. When studying workarounds and fitting work, it is tempting to study the situated use of the system to look for workarounds or other alternative systems and to judge those as superfluous if the system provides the same functionality in itself.

As we have seen in the previous sections of this thesis, misfit can present itself akin to the dystopian definition. The fact that the ERP system in a few cases produced what the workers believed was blatantly false information, or the ways in which users cannot query the information in the way they desire, is misfit according to such a definition. However, we have also seen other kinds of misfit that did not carry these connotations, such as the ways in which users communicate and transfer information outside the system – the users are clearly working around the system, but there is no real misfit between the system and the users in those specific usage scenarios.

Misfit is not inherently bad or undesirable, even though specific instances can be. Instances of misfit can lead to workarounds being created, but workarounds do not imply misfit. And lastly, misfit is generative, in that it spurs innovation and bricolage in the hands of the workers that are trying to compensate for the misfit. Misfit, then, can through its generative abilities lead to better end-states than had the misfit not existed in the first place.

3.3.2 Transparency

The workarounds and bricolages that were discovered had varying degrees of transparency. Transparency in this context refers to the level at which the workaround or bricolage is either recognizable to the user as a construct in itself, or seen as an object at all. Transparency is very well illustrated in Suchman's quote of Dreyfus:

"Consider the example [...] of the blind man's cane. We hand the blind man a cane and ask him to tell us what properties it has. After hefting and feeling it, he tells us that it is light, smooth, about three feet long, and so on [...]. But when the man starts to manipulate the cane, he loses his awareness of the cane itself; he is aware only of the curb (or whatever object the cane touches); or, if all is going well, he is not even aware

of that ... Precisely when it is most genuinely appropriated equipment becomes transparent. (Dreyfus 1991: 65)" (Suchman 2007, 73)

In the example of the blind man's cane, the cane becomes transparent to the blind man once he starts using it, and he is no longer aware of the cane, only the edge of the curb it touches. However, should the cane break or become maladjusted, it immediately loses its transparency and becomes opaque, in what Suchman calls a breakdown. This research has found that transparency is a very applicable attribute of workarounds.

Some workarounds are very intentional and can thereby be perceived as opaque to the end users, while other workarounds that are more embedded or casual can be those that have dissolved into transparency given time or distraction. Further, a workaround can become highly transparent because the end user has become habituated to it by regular use over time, or a workaround can be transparent because it simply is not perceived as a workaround by the end user.

Just as the primary tool itself will disappear into transparency when its use becomes sufficiently situated and integrated, so will the use of the workaround. It is apparent that when some workarounds and bricolages become regular practice to a big enough extent, the end users are not even aware that they are working around the system.

As an example, the locally developed system that has some form of structure or processing, such as spreadsheets or separate order books, are visible to the end user as an augmentation or addition; it stands in a superposition with the system. As such, the bricolage is opaque and the user has some awareness that she is actually duplicating records or neglecting to use functionality present in the ERP system. However, some other forms of workarounds such as e-mail, phone calls or more ad-lib'ed data structures does not give the end user a feeling of working around the system. The use of these workarounds are so embedded into the daily routines of the users, perhaps one can even describe them as second nature, that their use does not strike the users as having the traits of a workaround, and they can be said to be very transparent, even though they are not ideal or perhaps desired by the end user.

Transparency can vary between different usage scenarios even when the tools employed have similar properties. For example, in the case of Adam and Ben exchanging warehouse orders outside of the system, the workaround is fairly opaque to both users as they both acknowledge its existence readily and speak of its attributes as being outside the system. Conversely, Fabian's order-tracking spreadsheet is structurally not wholly unlike the attachments of Adam and Ben's communications, while still being much more transparent since he considers it to be totally embedded into his working processes. Fabian does not assume that the tool could not exist, because it has become intrinsic to his departments' nature of work.

The formal system itself can also have varying degrees of transparency. In the case of Helen's disregarded feedback to the developers, the shortcomings of the ERP system make it opaque and present, through a breakdown, as opposed to the workarounds that remain transparent, as they are still functioning, as far as she is concerned.

A very transparent workaround or bricolage combined with a very opaque formal system can also contribute to feelings of alienation and disconnectedness, which will be discussed in the next section on peripherality.

Further, the idea of transparency of workarounds is relevant because it allows us to look at workarounds with the same toolset as that we use when analyzing the primary tool or system itself, as the transparency will eventually blend the workaround into the formal system itself. The user can end up perceiving both the formal system and the workaround as the same system, since the transparency blends the workaround into the formal system. This has interesting implications for researchers and developers, who become forced to consider the workarounds – that can be unknown to them – as part of the formal system. This is for example apparent if the developers of the system change some attribute that will cause the workaround to experience a sudden breakdown, rendering it opaque to the end user.

The idea of transparent workarounds is also interesting for researchers because the users will talk about the workarounds implicitly when discussing the formal system. The feelings and experiences the user lends to the system might not actually come from the system, but from the situated use of the workaround, which in turn has become so transparent that the user no longer can tell the difference between it and the formal system. In this way, the formal system can have a range of traits attributed to it, which in reality should be attributed to the workaround.

To some extent we can say that transparency can be a measure of the embeddedness of a workaround into a given system or work context, simply because the more integrated a workaround becomes into a work process, the more transparent it will become.

The measure and discussion of the transparency of both the workaround and the formal system itself then seem to have some merit. Understanding the transparency of workarounds lets us understand to what extent a workaround is embedded in the work process, and how it must or might be paid attention to when studying the formal system. Understanding how transparent the workarounds are will enable us to differentiate them from the formal system when studying situated use. And ultimately, the more transparent the workaround is, the bigger the breakdown will be if the formal system should suddenly change or stop supporting the workaround. Developers should therefore assume and understand the existence of workarounds bricolages, and might to some

extent plan how the consequences of breakdowns can be alleviated. Consequently, the more opaque and present a system is, the lesser the consequences of a breakdown might be.

Further, similar tools can have different levels of transparency for different users, thus necessitating the understanding of a variety of possible use cases for workarounds and bricolages in different settings.

Lastly, a series of very transparent workarounds combined with very opaque formal systems might also contribute to a user's perception of not being part of the formal system; an effect that we shall elaborate on now.

3.3.3 Peripherality

The tools, workarounds, and bricolages, that are created by the users are generally regarded as appropriate and fitting for the purposes that they are created to serve. While they might not be considered perfect, they are held in high regard because they are created with the intent of serving the purposes and needs that are close to their maker: They were brought about and are maintained as creations for and by those who are close to it. The same cannot be said for the formal system that the users interact with. There are strong correlations in the analysis results between discussions of the ERP system and the expression of a feeling of *peripherality*.

Indeed, it might seem that the users sometimes perceive themselves as peripheral to the purposes and business processes of the ERP system. In the first drafts of this thesis, this effect was termed *externality*, simply from the sheer firmness with which the interview subjects spoke about it. As we shall see, peripherality to the ERP system tells of how the end users believe themselves to not be the prime target audience for the system, or of the system not being designed for use by them. The term seeks to encompass the observed distance between the end users and the formal system and its structures, prevalent from its identity and implementation to how it is (not) regarded as part of the end users' jobs. As such, the end users might not interpret the ERP system as a tool to support their primary work, but rather solely a system of accountability.

For example, the analysis has shown that the end users constantly talk of a "*them*", referring to those who are in charge of the development of the system or those who read reports based on the data that is entered. As such, the users do not have any clear idea or vision about the ownership of the development processes of the ERP system, nor do they seem to know what parts of the organization, if any, is actually extracting or interpreting the data that they are entering into the system. This effect is interpreted as a marked gap between the end users and the system, leading the end users to believe that the system is not developed for them while they simultaneously do not know who benefits from using the system. This will obviously hinder development of serious commitment to the system as the end users cannot see any particular reason, beyond the obvious

fact that they are requested to use it, to commit to using the system as their primary tool in the tasks that it supports.

Fabian is quite direct when he tells that *"nobody in central management works for those of us out here [in the organization]."* Ben, slightly more restricted, explains that *"it's important that we're aware that [the ERP system] is a tool for the 'higher ups' to take out what they want."* Dina adds, *"I guess all of those strange codes are there so that others can look at the ledgers at a very detailed level."* Adam is a bit more colourful with *"we're working while banging our heads against the wall."* Fairly universally, these statements display a quite clear feeling of peripherality to the ERP system; the users do not feel that the system was designed for them or with them in mind – the job they're doing in the ERP system is not related to their primary function or made to serve themselves; it is an action performed to provide data for someone else or for some other purpose unbeknownst to them.

Ben comments that after requesting a set of products and sealing the deal for a purchase from an external supplier, he has to punch the details of the purchase into the ERP system – a process he calls *"getting a number"* from the system, referring to the consecutive numbers that the ERP system generates for the purchase orders. This number is not given to the supplier, but kept in internal documents together with the internal number discussed previously. When the invoice from the supplier arrives, the invoice is not automatically paid because it is only marked with the internal reference number and not the purchase order number. Ben has to look up the reference number from the archive, find the purchase order number and list them in the system every time.

Dina remarks that *"I'm sure there's a very good reason why it is the way it is, but I'm having problems seeing that when I'm dealing with supplies that just /have/ to get dealt with."* She continues, *"[the ERP system] is not designed for a station with two people"*. Adam, the warehouse manager, points out that in the hectic period in which the mobile unit his warehouse services is present, he doesn't have time to perform transactions in the ERP system because he *"has to work"*.

Fabian aptly comments, *"there are no restrictions in the spreadsheet application."* Further, he says, *"I don't know what's going on in the ERP system."* The feeling, again, is that the department is peripheral to the purpose of the ERP system, while the locally developed system is the correct or proper system, which serves the present and situated needs, whereas the formal system serves someone else.

Eric comments on the lack of understanding of the true purpose of the ERP system, stating that there are multiple other links and events firing behind-the-scenes that the user might not be aware of:

“When you begin a process in [the ERP system], for example the purchasing process, then it’s not only that process you’re starting, but you’re also starting economical processes like commitment processes, warehouse processes and those sorts of things. If we’d only had the purchase process, and only going to focus on purchasing and disregarding commitments or invoicing or any of those things, then it would have been very simple, but there are many other concerns to account for that make things more complicated.”

If we choose to see Eric in his role as a trainer in the ERP system as a spokesperson of upper management, it is not hard to see where he would put the blame for failure and rejection of the ERP system among the departments:

“Locally it is mostly about organization. I think that many departments are terribly bad at organizing themselves. They are organizing themselves into trouble, quite simply. That’s my opinion. They organize themselves awkwardly, and it becomes more awkward to do things.”

While being quite blunt, his statement does corroborate with some findings. Look for example at the case in Dina’s department: She is both physically and organizationally located away from the end users she is supposed to serve. It is not outrageous to imagine that the process would be much easier to adhere to, had she been closer to where the action takes place.

As a further example, one of the end users interviewed, Helen, is found to keep track of all the purchase orders she registers in the ERP system on a paper notepad she keeps beside her computer. Upon question about what the purchaser notes down in her paper-based journal upon every purchase, Helen informs that the journal contains the order date, name of the supplier and price. Only after the next question does she remember that the journal also carries the purchase order number generated by the ERP system. It is clear that the purchase order number, which is the primary key of the purchase order generated by the ERP system, is not even considered to be remotely important or noteworthy by the end users - the key for them is the combination of date, supplier name and price, not purchase order number that the ERP system insists on using. We can anyway see that the end users build around this inconvenience by making their own constructs to compensate for the lacks felt in the system. As Helen says:

“<What are you noting in your notepad?>

Date, purchase, and price. Or what is it, supplier and price.

<And purchase order number?>

And purchase order number, of course. One manages to keep some track, then. But ideally it should not have been necessary – the notepad should not be necessary because this should go by itself. And if it's an exception, let's say the quantity or invoice is wrong, then I should immediately see that – 'here's the purchase order tied to that invoice, but the price is a little bit wrong and you need to correct it.' Okay, fine. But when I don't have – when invoices arrive, then, as if they were going to – as if they were going to be post-hoc processed, even though they are registered in advance, then it becomes a bit – you need to keep cool, you need to have a structure for what comes in. So... it's a lot of [one's] own structure... a lot is dependent on one's own structure, and your skill, then."

While an obvious example of a workaround, as has already been discussed, this behavior is also interesting to observe as an indicator of some form of peripherality between the user, and the system and its structures. Since the users do not know who either develops or extracts information from the system, their commitment to the system and its structures are challenged and their belief in this construct diminished.

As a further example of this effect, it is striking that most subjects talk about the ERP system by referencing it by its brand name acronym – for example as *"it will be registered in [ERP]"* (Graham) – and not by the functionality that the system provides. The units in the organization that have constructed their own alternate lists, outside of the ERP system, rarely talk about these lists by anything but their functional name: They do not say, *"I will put it in Excel"* or *"it is then inserted into the binder"*; rather, they say that *"I entered it into the our purchase order list"*, referring to the Excel spreadsheet, or *"I filed a copy in our invoice archive"* referring to the physical binder on the shelf by the workstation. As such, it is apparent that the end users consider their own local systems to represent a more relevant system, or a system that is more involved in their own processes than the ERP system, which is perceived as peripheral to their central business processes.

It is prudent to note that the way the end users constantly reference the ERP system by its brand name might also be grounded in the fact that the ERP system is so heavily branded and of such deviating visual appearance that it makes its own presence very well known to the end user, and therefore visually and systematically represents a frame of mind of its own. However, this seems less likely, considered together with the other observed indicators of peripherality and distancing.

As another example of a more organization-wide case of ERP system peripherality, it is interesting that one end user remarked that all of the reports submitted to upper management are extracted from the ERP system by the financial controllers each organization, manually, and then submitted to upper management outside of the ERP system:

"Hm... no, like that... no, yes, no, there are nobody who... it's rather what I take out of it. And none of the reports I deliver are taken out directly – none are – none are a picture of [the ERP system]. Everything is what I've taken out and my own generated image.

<That you've processed in some way?>

That I have processed, yes." (Helen)

Even upper management, at this stage, does not employ the ERP system as its primary tool. An organization-wide system has been introduced, and upper management elects to work around it? Perhaps they do not trust the uninterpreted data coming straight from the system. Regardless, in such a case, it is easy to see how the departments of the organization eventually distance themselves from the formal system and elect to see their own bricolages as the core of their own primary work.

This is not to say that the ERP system holds no respect or is in no regard among the users; on the contrary, the users are all aware that it is the version of reality that is kept in the ERP system that is the one that they will be held accountable for, and that the ERP system will be available for investigation by others for years to come. As such, it can be imagined that the end users see the ERP system as more of a tool for accountability and supervision than they do as a workhorse tool for their daily tasks – a perspective which will be discussed later.

Regardless, the users remain detached from the formalities of the ERP system, and they do not consider it as a tool for them – it is a tool for someone else. Having surveyed users from several different departments of the organization, it is challenging to say exactly who this someone else might eventually be. The users, at any rate, do put stock by the ERP system, but they do not see it as their primary system.

A perception of peripherality of the formal system can be a major contributor to the development of workarounds. As we have seen earlier, many features that are available in the formal system are replicated in the workarounds and bricolages that appear in different parts of the organization, but since the users feel no need to commit to more than the absolute minimum of what the ERP system requires – just to *"get a number"* so that the invoice can be paid – they seek to establish the structures they require in their own bricolages. In this context, the ERP system becomes opaque, almost as an obstruction, while the bricolage is the truly integrated tool for their primary work, one that they can trust and care for, simply because it is not peripheral to their job.

3.3.4 Detective Work and Constructed Rigidity

The purchase order regime is as already discussed established to enable a central oversight over the commitment plan, a central list of resources that have been assigned and committed to a certain activity but not yet spent or paid.

As multiple subjects discuss, the main process is not always complied with – in which the purchaser registers a commitment in the ERP system before the invoice arrives, because of errors and difficulties observed when processing incoming invoices. Upper management wants the departments to register as many purchases as possible before the invoice arrives, but purchases performed according to this model are more prone to errors or problems than purchases performed using the post-hoc registration method. When following the main process of purchasing, the purchaser has to create the purchase order in the ERP system before the purchase order is sent to the supplier. To do this, the purchaser needs to know the correct values for a host of variables that might change during the purchase process such as product descriptions, pricing, quantity, delivery terms and shipping expenses. These variables are often subject to change, and if that were to happen, the invoice will not be automatically linked to the purchase order. Further, the suppliers are prone to omit the purchase order number reference from the invoice. If any of this should happen, the invoice will appear in the inbox of the ERP system in a variety of states depending on the nature of the problem. The purchaser is then required to correct the problem by modifying either the purchase order – if the invoice is correct – or by modifying the link between the two if they somehow failed to link.

Several of the subjects refer to the process of recovering from such situations as "*detective work*". Helen makes apparent that while the mandatory purchasing course in the ERP system gave the purchasers a functioning insight into a problem-free process of purchasing, there was little or no training in the treatment of problem situations or purchase processes that by necessity deviates from the planned course of action. As such, when the plan of the purchase process breaks down, the value of the training and operation manuals are virtually nullified and the operator is left to explore the functionality of the system on his own, trying to solve the problem by using a combination of exploration, creativity and trial-and-error methods. The ERP system renders no help as the system is not aware of the desires the end users, and the end user lacks the necessary vocabulary to explain to the ERP system what actions are to be taken.

Relating to the process of breakdowns, it has also become apparent that the combination of very light training and less-than-intended time spent in the ERP system has led to a situation where the end users are generally comfortable performing their tasks according to step lists or other

instructions, but on the other hand, have little or no knowledge about how to deal with breakdowns or use cases that fall outside of the predefined process plan.

Specifically, breakdowns that easily could have been handled by a more-skilled user, or frustration stemming from receiving work objects that is not the end user's responsibility seems to be a major point of frustration for the users.

For example, when the organization receives an invoice that does not carry a purchase order number or the identity of the purchaser, the central accounting department might attempt to assign the invoice based on fleeting or more subjectively interpreted data such as the delivery address, projects mentioned, or even the bill of materials itself. Further, a supplier that lacks the proper purchase order references for the invoice might re-use previous purchase order references or even names of other purchasers, so that the invoice will be not be rejected by the central accounting department.

This leads to situations where the purchasers will be assigned invoices that they have no knowledge of or relation to, and that it is not in their mandate to approve. At this point, it is also important to note that the ERP system itself features no way of rejecting or returning an invoice. An invoice can seemingly only be linked to a purchase order and accepted, forwarded, or if it's wrongfully invoiced, linked to a credit note. The latter option, however, is only valid for invoices where the content of the invoice itself is not valid - for example, if the invoice is based on inaccurate data, the goods were returned, services were cancelled etc. As such, when the purchaser receives an invoice that probably has validity - it looks like it is related to an actual activity or goods acquisition - the purchaser has no way to deal with the invoice but to track down the intended recipient in person and forward the work object to her inbox. The work is generally characterised as being undesired by the users, since they are neither part of the cause nor going to be part of the solution by actually accepting the invoice. Further, the work is time-consuming and resource demanding since it requires the end users to venture into unknown territory by trying to dig up names and numbers of people outside their own department. The users cannot reject this work, because the system does not offer any technical way of doing so, and they must deal with it, because the existence of an unpaid invoice will reflect badly on the purchaser's organization. The subjects of the study referred to this kind of work as *detective work*. This kind of work is characterized by the user being unaware of both the object of work – “to whom shall I forward this data?” – and the method of work – “how do I go about figuring out who it should go to?” – imposing a double-bound situation on the user.

The interesting thing to note here is that none of this would be a problem to the end users if the end users were to register all of the purchase order post-hoc. In such a scenario, the end users would already have the correct properties of all the variables, and would be able to follow the

purchase order to completion instead of leaving it in a semi-available state that might or might not work as the process progresses. However, they should not do so, because upper management has decided that more than half of purchase orders should be registered before the purchases are performed, a decision that has been implemented into the system. The various workarounds thus become products of the rigidity of the formal system, and this rigidity is not given for technical or juridical reasons: The system is rigid because it has been constructed so, the formal system carries a *constructed rigidity*.

The notion of constructed rigidity can be useful in the discussion of workarounds. Especially, this research shows that a system with a high degree of constructed rigidity might have a higher number of workarounds created around it, or at least, that a system where some of the workarounds are created to overcome the limitations of this construct. Essential to the notion of constructed rigidity is the fact that the limitations that impose the rigidity on the system do not stem from any power external to the organization, such as a law or the common knowledge of a best practice, but has been created by the organization for reasons and purposes that the end users don't necessarily agree with.

3.3.5 Trusting the Bricolage

The users have little or no knowledge about the processes going on behind-the-scenes in the ERP system. While the users still provide the data that is required by the ERP system in various processes, they report being unaware of what the purposes of these data are, or who will use it. This lack of insight leads to a loss of trust, which is contrasted with that found when talking about the workarounds and bricolages.

Textualization (Zuboff 1988, 315), the interpretation and codification of work practices, often form the basis of the automation of work. In the development and introduction of a formal system in a big organization, the textualization might take place close to those responsible for the introduction or integration of the formal system or it can already be completed even before the system is acquired – but it seldom takes place close to the end user of the system. As such, the interpretation of how work goes about that will become embedded into the formal system might or might not represent the optimal, desired or currently practiced routines among the performers of the work.

Formal tools use conventions that might be unknown to the end user, while open-ended tools like spreadsheet applications allow the development of a tool that will follow the thought pattern of the user – thus the applications that the users employ in the creation of tools and structures appear easier to understand to them, because they inherently understand the business

process that they have created themselves. In other words, a general office support application will seem easy to use because the structure developed in it is inherently understood by its creator.

As observed in the departments, the users are often wary of using the ERP system because they do not know in what fashion it treats the data or what process it follows, or put more simply, how it works. The textualization of the process embedded in the system is hidden from the user. The insecurity and perhaps scepticism caused by this evokes the user to create her own solution by performing her own codification and textualization, limited by the tools and methods available to her as a bricoleur, thereby putting down her own concept of the idealization of the work. In the process of creating their own processes and tools, the bricoleurs will to some extent perform their own textualization; their own interpretation and codification of how the work is performed, as observed, felt, lived and concluded by the workers themselves. Trustworthy, secure and explicit, the bricolage reveals itself as a product of both the user and the formal system – as desired by the former and as provoked by the latter. The bricolage grows to meet the needs that are unfulfilled by the formal system, or as proposed by Schmidt and Bannon:

“That is, the system should make the underlying model accessible to users and, indeed, support users in interpreting the procedure, evaluate its rationale and implications. It should support users in applying and adapting the model to the situation at hand. It should allow users to tamper with the way it is instantiated in the current situation, execute it or circumvent it, etc. The system should even support users in modifying the underlying model and creating new models in accordance with the changing organizational realities and needs.” (Schmidt and Bannon 1992, 20)

In this way, the users place more trust in their own bricolage because they know how the tools work and they can appreciate and explain the purpose of every single feature, whilst the formal system provides no insight and the users do not know why certain actions need to be taken, how the data is processed and where it is sent.

The positioning of the workflow management provided by the ERP system as internal to the formal system is then questioned. Not only does it prevent insight into the workings of the system, but it also enables the system to strictly enforce its own limitations on how and what is to be done.

“Of course, those who criticise workflow systems are often precisely objecting to the internal positioning of the system within the work. It is this, some would argue, that allows workflow systems to overly constrain the work by imposing some process model or theory of interaction on it. If a workflow system were external to the work and did not directly control the availability of resources for the work, then perhaps more flexible

support for cooperative work could be offered through workflow.” (Bowers, Button, and Sharrock 1995, 53)

The creation of tools that are external to the formal and accountable data store, the ERP system, then represents a materialization of the desire for flexibility, insight and trustworthiness.

3.4 Accountability and Organizational Ordering

We have so far entertained a discussion on some actual embodiments of workarounds and findings that can be described as either a set of characteristics or attributes of workarounds. The study also suggests interesting relations between the use of workarounds and factors relating to the use of the computing systems as methods of accountability and organizational ordering, which we will detail now.

Many studies have been made into the consequences for both organization and the individual worker, of the introduction of computing systems in general (Zuboff 1988; Bowers, Button, and Sharrock 1995; Thomas 1994), and ERP systems specifically (Grabot, Mayère, and Bazet 2008; Hanseth and Braa 1998).

Barley (1986) explores the relationship between organization and technology, and while the technology certainly can be seen to shape organization, the effect is not necessarily predictable or replicable across different implementations. As such, organizational structure cannot be taken as an a priori given of the human involvement in the process, but must rather be viewed as emergent from implementation and use. While plausible, some credit for the resulting structure must also go to the implicit structural payload of the technology – as such, the technology both carries and enables structuring, or *“appears to require a synthetic view of structure as both a product of and a constraint on human endeavor”* (Barley 1986, 79). In particular, Barley found that the introduction of identical CT scanners in different organizations brought about two similar structuring processes that still led to different forms of organization – a result that might be counterintuitive, since the developers often assume that the product will lead to similar situations across implementations, in that *“assumptions about how a product can or should be used are often designed into the product itself, and [can] be very influential in how the buyer organization structures use of the product”* (Thomas 1994, 190).

This underscores the relevance of studying how technologies occasion structuring and accountability in multiple organizations, as no two organizations are ensured similar outcomes. To both observers and users, it is clear that the introduction of a new ERP system will necessitate changes in the daily work routines of the users of the system. This discussion will take on the attempt of interpreting the situation from the users perspective, building on the collected data and the findings presented so far.

Through the effect of perceived peripherality of the system, the end users are of the clear opinion that the formal system is owned and operated by someone else and in the interest of someone else. From this standpoint, even the end users can envision the system as a system of accountability and organizational ordering. This is partly based in the users supposing that this must be the intent of these others that the system come from, as they fail to see the system as a way for supporting their primary work.

Recall that the organization used to have several smaller-scale systems for many business processes like finances and HR, as discussed in chapter 2.2. These systems have been shut down in favor of the newer organization-wide ERP system. It is possible that some of the departments were using the same system as the one that was introduced organization-wide, but for all the others, the introduction of the new system along with its clear description of new structures and roles came with great consequences of organizational ordering. This was not only evident on the local level where the change of ERP system obviously entailed a change of work practices for the end users, but also on the macro level where entire departments were restructured or shut down – for example, the accounting department of the organization is now composed of employees from all parts of the organization.

The system can also be envisioned as a system of accountability. For example, the formal ERP system makes it much easier for managers and other parts of the organization to observe the detailed work of each end user; both the results of the work and in some cases the work as it appears in progress (Ciborra 1997). It is not given that every user finds this desirable. As such, the use of the formal system might be reduced to the absolute minimum as demanded by either the system or the organization, whereupon the bricolage will be entrusted the task of keeping the entire overview (Ciborra and Patriotta 1996). In this way, the workaround becomes a way of keeping back some pieces of information from the overlying system of accountability. However, the desire to keep certain pieces of information outside the system is not the only reason why the accountability offered through the system is worked around by the users. Given the amount of insight that both local managers and upper management can get by employing this system, the system is also open to interpretation in the form of a panopticon.

The panopticon, as conceived by the philosopher Jeremy Bentham and presented by Zuboff (1988, 319), is an architectural construct designed with the intention of providing an unrestricted insight into the activities taking place within the confines of a complex, be it a factory, workplace, asylum or prison. The true power of the panopticon, however, lies not in the oversight provided to the central agency, but in the behavioral changes imposed on the subjects by the awareness of the existence of the panopticon. The textualization and informing of the modern workplace has

removed any argument as to why management in any organization cannot be given such insight into the processes that take place in the computer systems, save for the opinions of privacy. Thus, the connectedness of and the computational resources provided by modern computing platforms opens up the data sources to the possibility of continuous and relentless scrutiny; any data set carries the possibility of being inspected at any time by many different roles and parts of the organization.

Choosing to interpret the ERP system as a form of the panopticon, we not only see how the central agency – which the users assume will be upper management – now has unrestricted insight into the workings and goings on of the organization, but also of how the knowledge of the existence of the panopticon shapes and changes the work of the end users. This puts the formal system and the agencies behind it in a position of power over the users. While the panopticon in its traditional and architectural form is obvious and obtrusive, the oversight provided in the ERP system is less intrusive and more discreet, as the system is surely not built for the single purpose of providing oversight, but the effect is nevertheless present. Among the many consequences of such a situation, two possible outcomes are especially relevant in this case.

Particularly, Orlikowski (1996) found that managerial insight into micro-level processes can bring about two differing outcomes, perhaps determined by the mindset of the end users. After the introduction of a performance monitoring system, some users expressed concerns of surveillance, while other users found the insight provided them a way to show off for and impress the managers. In this way, the users sustain a digital image of their looking-glass self (Cooley 1964; in Baumer, Sueyoshi, and Tomlinson 2011); a representation that they not only maintain but perhaps even cultivate. In respect to the present study, it is conceivable that users who are unable to do so by showing exceptional prowess in the employment of the ERP system choose to create their own tools in which they can show off and gather managerial praise: If one cannot excel in the set playing field, the formal system, then create one's own playing field in which one can master.

Extrapolating from Orlikowski's findings, we see how the existence of the formal system can spur the creation of workarounds not only from concerns of insight and accountability, but also from the pure desire to master or excel at a given task. A perceived feeling of competencial inadequacy is persistent in the data gathered for this study, and the notion of employing the at-hand tools in a show of proficiency is especially central to the notion of the bricolage. As such, it is very interesting to note that the upper management level chose not to extract their reports from the ERP system, but rather chose to enlist the help of all the representatives from all departments in the organization when preparing reports, and accepting only the reports structured, explained and annotated by the representatives from the departments themselves. While not a direct message of distrust in the ERP system, it underscores a desire for a level of analysis and deciphering not presented by the ERP

system in its top-down-views. The end users, then, get a reward from their careful bricolage – they get to show it off to upper management. The ERP system, on the other hand, treats every department and every user the same, and does not allow anyone to create a report that will be favored because it includes dazzling visualizations or insightful interpretations.

Not only does the ERP system treat every employee the same, but it also makes it harder for managers to track performance and quality of the products of their employees. A rigid compliance with the formal system will result in the manager receiving only the notifications prescribed by the plan, while a deviation from the plan will generate few if any messages. In other words, for the ERP system, failure is never an option, as the constructed plan cannot be deviated from. This is easily seen in how many processes imposed by the system are not optional to the users – they can only respond by giving the system what it requests, as discussed in chapter 3.3.4. This not only emphasizes the formal systems role as a system of control and ordering, but it nearly relieves the managers from the responsibility of managing their subordinates. Zuboff (1985) argues that this is an effect prevalent in computerized systems, grounded in the nature of the presentation of such abstracted data. She holds that the resolution to this dilemma is the manager's trust in the diligence of the worker, as it is only in the interaction between system and user that efficiency is ensured:

“In a conventional environment it is relatively easy for a manger to determine that a worker has not properly repaired a boiler (it continues to malfunction) or failed to type a document properly (it is full of errors). But how does a manger determine that an employee failed to respond to some element in the data? How does a manager evaluate the possibility of missed opportunities to learn more about the business or improve operations in some way? In the final analysis it is only the employee's skill and commitment that can ensure that intellectual effort will be exerted and that opportunities made available by an informing technology will be exploited.” (Zuboff 1985, 16)

In the light of how managerial oversight is lost in the formal system, it becomes not only the end users desire to create alternate data structures to track and visualize how work flows when it is misrepresented by the formal system, but it also becomes the managers desire to get insight into these creations as they are the primary way in which users track, order and coordinate their work. Deviation from plan is never considered an option by the formal system, and as such, it does not represent it in any way. Since the workarounds and bricolages that have been created by the end users represent the end users own interpretation of how work unfolds, these constructs are much better suited to acknowledge the actual status of any given task or process. The workarounds and bricolages benefit from the ability to acknowledge that not all work unfolds in the same way, and

since they are adaptable, they can continue to change as new ways of working come about. Additionally, since they are created locally, they have the ability to decide who gets access to these constructs, and thereby counteract the panoptic notions of the formal system.

Does this mean that the workarounds and bricolages can be seen as an attempt to resist the ordering implicit in the formal system? Obviously, not all workarounds and bricolages exist solely for this reason, yet some do. Rodden and Blair (1991) presents a classification scheme for CSCW systems, arguing that there exists two main styles of control offered by such systems:

“In systems which provide explicit control users may both view and tailor group interaction and cooperation. In contrast, systems exhibiting implicit control provide no techniques for representing or coordinating group interaction. These systems dictate cooperation by the styles of interaction they allow.” (Rodden and Blair 1991, 57)

Rodden and Blair claim that office procedure systems, systems that support *“tasks performed within an office in terms of the combined effect of a number of small sub-tasks or procedures”* (Rodden and Blair 1991, 57) offer explicit control in that they afford a structure that allows the *“representation and editing of control information”* (ibid., 58). However, this interpretation does not hold in the situated use of an ERP system like the current case, in which the end users are not able to either view or change how the system works – the classification is only valid from the vantage point of the system developers that are not part of the actual use of the system. This lets us understand how the developers and owners of the system fail to see the need for existence of workarounds in the organization – to them, the system offers explicit control, since they can tune the system to match their desires. The users, on the other hand, are subjected to the implicit control of the system; in essence, they are under the control of the developers and the upper management that regulates the use of the system. The workarounds and the bricolages can then be seen as ways of resisting the control, or ordering, implicit in the formal system. Further, the users are then enabled to perform textualization of their own work, as opposed to letting the interpretation of work practices be performed by another party.

Kallinikos (2004) argues that the models imposed onto an organization by the introduction of an ERP system constrains the individual worker freedom to choose how the work is to be interpreted and performed:

“The transactional mechanics which ERP packages bring about may thus block exploration of alternative ways of perceiving and acting upon reality and by extension organizational development and innovation (March, 1991). The opportunity to experiment, improvise and rehearse with alternative ways of perceiving and acting upon the world thus presupposes forms of human involvement that are sharply distinguished

from human behaviour as rule-following. Such forms of human involvement, often revealed in improvisation (Ciborra, 1999; Weick, 1979a, 1993) and play (Bateson, 1972; Kallinikos, 1996; March, 1976), collapse the distinction between general and procedural knowledge, knowing what and knowing how.” (Kallinikos 2004, 20)

Kallinikos argues that the structure inflicted by any ERP system separates the object of the work, or what is seen as data to the ERP system, from the performance, structures and routines that are given through the combination of the worker and the system. Contrast this to the bricolage observed earlier, where the object and product of the work was not only the data but also the structure, framework, body and content of the data being produced: The work is, to the ERP system, only about data, not an individuals interpretation or opinion about it. Working around the ERP system then becomes an act of adding one’s own interpretation of the data to it. This is apparent in such workarounds as Fabian’s spreadsheet, which tracks not only purchases as abstracted objects, but as concrete items like fuel, provisions etc., a practice that *"models the natural work practices of [the] individuals"* (Coovert and Thompson 2001, 39). Coovert and Thompson argue that this is essential to the success of a system, and teamware in particular.

As previously touched upon, the ERP system was introduced alongside an organizational restructuring that mostly affected those users who performed tasks that would now be supported by the ERP system, such as accountants and HR workers. A restructuring like this will inevitably be met with resistance from some employees. Barley (1986) points out how the formal organization is tightly wound into the rules set into systems through in how they delimit action and interaction, and how technology and organization can bring about the structuring of each other:

“Just as scripts can be conceived of as behavioral grammars that shape instances of action and interaction, what we traditionally call formal organization can be viewed as the grammar of a set of scripts. [...] The link between action and formal structure can be visualized as a chain of successive encodings that abstract, first, from instances of action and interaction to properties of scripts and, then, from scripts to properties of formal organization. [...] Thus to occasion the structuring of organizations, technologies must first disturb or confirm ingrained patterns of action to reformulate or ratify scripts, which, in turn, delimit the organization’s institutional structure.” (Barley 1986, 83-84)

Given this view, the refusal to comply with the rules given by the ERP system through the upkeep of workarounds and bricolages can be seen as a resistance of the entire restructuring process. This resonates with the description of the developing effects of peripherality described earlier.

At some levels, the end users become organizational bricoleurs, deciding not only how they go about their own work but also how interaction takes place and how cooperative work unfolds.

As we have seen in the data, successful bricolages can become integral to the work and achieve widespread use within a department, and by extension, the workers can provide a bottom-up restructuring of the organization as their tools that imply a certain order of work become more successful and accepted. The structuring imposed by the formal system is deterministic and difficult to challenge. The bricolage and its adoption, on the other hand, is opportunistic and becomes the democratic voice of the users – the bricolage represents the way the users desire to go about work, and how these teams should coordinate and structure their tasks. Since the bricolages are the creations of the workers, they are subject to their adaption and development, and not that of any upper management. This is the organizational bricoleur at work.

Not all workarounds and bricolages are candidates for contributing to organizational bricolage. Dina's paper archive of invoices filed in two binders both chronologically and alphabetically, might not carry strong structural consequences. Fabian's spreadsheet where he tracks all purchases might. His spreadsheet is used to generate reports of purchase activities, and also as a tool for cooperation with the other employees who perform similar functions as him. The ways in which his organization perceives that this tool adds value to the processes might have consequences for how future structuring is performed, or what functions are considered important or not. A respected bricoleur can be asked for organizational recommendations by local management, under the assumption that his insight into the improvement of work through the introduction of such a bricolage has given him an awareness of how the organization could be better structured.

Schmidt (1999) touches upon how artifacts are used as structuring devices through the process they convey in their design, for example by reminding users of certain actions by offering empty spaces in tables and checklists. The creation of one's own tools of work can therefore be a rejection of the structure or the imposed tools, like those of the ERP system, allowing the end users to set their own agenda, especially for coordinated activities.

Perhaps it is too often assumed that it is up to the organization to change, and not the system. Take the very name of "technologies of accountability and organizational ordering" – it suggests that it is the technology that is to order the organization, and not the other way around: Grabot, Mayère, and Bazet (2008) christen their publication *"ERP systems and organizational change"*, Orlikowski (1995) chose the title *"[...] Organizational change around groupware technology"* and Ciborra (1997) asks *"what does groupware mean for the organizations hosting it?"* An example taken a bit too far, perhaps, but nevertheless: Might it be that the organizations expect too much of the system that is implemented; that it is not the organization that should bear all the burden of change, but rather how change on both accounts can grasp the full power of the system?

An adaption on both parts would better be suited to leverage the true power of informing (Zuboff 1988, 1985), to which we shall return later.

Regardless of what the optimal configuration could be, it is clear that the ERP system studied can be interpreted as both an accountability and ordering system, an effect that the users perceive as an opportunity, with some reluctance. We can also see that the users' perception of the system as a system both of accountability and ordering might give rise to the existence of several forms of workarounds present. We have seen that users can use workarounds to keep some information off-the-record, and also how the bricolage can be seen as be an attempt at resisting the structuring imposed by the formal system – and even how it can be a way of introducing a new structure.

3.5 Workarounds, Bricolages and their Characteristics

The definitions "*fitting, augmenting and working around*" coined by Gasser (1986) in themselves imply or presuppose an inadequacy or other shortcoming of the computer system. They are all problem-oriented, as discussed in chapter 1.1. It can be conceived, at some levels, that this kind of appropriation work should not be considered in a negative or pessimistic way, but should rather be viewed as a natural extension of situated work.

Let us then take the example of an ironing board, a common household object to which most of us have some sort of relation. When one acquires an ironing board it is functionally complete, in the sense that it has all the features necessary to fulfil its purpose. Still, when it comes, collapsed, straight out of the cupboard or even the plastic wrap it was shipped in, the ironing board is in no fit state for comfortable or ergonomic use. True enough, it is possible to use it in a position a few centimetres above the floor, or one can place it in some sort of creative composition involving chairs, tables or able-bodied assistants. No, the person wishing to have her shirts ironed needs to depress the lever releasing the legs of the ironing board and extend them to a comfortable working position, perhaps adjusting it once or twice more before the final comfortable arrangement can be found.

In this situation, are we able to claim that the fitting work that was being performed was a result of some inherent misfit of the ironing board? No, we will respond quite naturally, it is quite obvious that one has to expand the legs of the ironing board before it can be used. Yet, the ironing board in its extended state is no more powered on than it was while it was in storage, and it is still composed of essentially the same parts and components, only now in a slightly different arrangement than earlier. Then can we claim that there was something wrong with the ironing board; that it was in some sort of problematic state that is now fixed? No, any seasoned ironer could be able to examine the board before it was unfolded and would agree that it showed no signs or

symptoms of fault and that it was in no unnatural state that would somehow prevent it from working.

The purpose of this metaphor is to point out that the work we have so far observed, which can be interpreted as misfit adjustments in fact can be quite successfully seen as a natural and even essential to its use; that it might not even be spoken of as appropriation work: As noted by Berg (1997, 151), such fitting work is not to be seen as work to appropriate or repair the state of the software into some other more desirable state, because appropriation suggests the existence of some optimal form or configuration, while the point here is that no special or singular configuration can be seen as wholly superior – superiority here is given by the object's inherent adaptability and flexibility with no preference for ideal or proper state.

As Suchman (2007) so eloquently puts it, it becomes not a question of when procedure is deviated away from, but when the situated action is seen as being carried out in accordance with a rule or law, or a set thereof (Suchman 2007, 194). The action is no longer to the letter of the law, but in the spirit of the law, serving as a basis for interpretation on the part of the actor. In this view, the procedure no longer requires strict compliance, but provides a norm for what the intent and purpose of the work should be. This presupposes the worker's ability to consciously deviate from the plan when interpreting it.

Indeed, to understand the nature and scope of a workaround – and even if it is, in fact, a workaround – one must not only consider the actual use of the workaround but also the intent of the system that is being worked around, and the intent of the workaround. Given the perspective of situated use, some actions cannot be workarounds as they represent the interpretation and adaption performed by the worker for the situation at hand. A hand-written list of purchases, suppliers, prices and purchase order numbers might constitute a substantial workaround – while a purchase order number jotted down on a post-it to give to a colleague over lunch might not. But in essence, what is the difference?

To get an impression of what the difference between these two examples might be, we can interpret their intentions vis-à-vis the notion of situated use. Scribbling a reference number on a piece of paper, in this light, is not a contradictory to the intents of the system, but rather an attempt at coordinating some piece of information between multiple systems or multiple workers in a given setting at a certain point in time. Conversely, the upkeep of a large structure of information outside of the formal system might be a workaround, since this is the replication of the intents of the system, beyond just adaptation and interpretation work.

Hopefully, it is non-trivial to tell when a workaround is. The existence of a workaround depends on our interpretation of the work situation at hand – if the formal system is to be

understood as a plan for action, then almost all situated use becomes a workaround; if the formal script is only to provide a spirit to act within in the situated use, then the playing field suddenly opens up and much more use can be seen as a natural part of the process.

We can see, then, that the misfit is indeed generative in that it spurs the creation of all these novel ways of going about work. Seeing misfit as generative, by extension so are also the workarounds and bricolages discussed herein. The understanding and discussion of these generative powers can become very beneficial to both the understanding of and the development for cooperative work arrangements. This is the crux of the development of a typology of workarounds.

Not only can misfit be generative in creating workarounds and bricolages, but its power can also extend further, altering the very work practices and structure of the organization in which it is implemented:

"[W]hen slippages persist, they become replicated patterns whose contours depart, perhaps ever so slightly, from former practice. Eventually, changed patterns of action reconfigure the setting's institutional structure by entering the stock of everyday knowledge about "the way things are" [...]." (Barley 1986)

Misfit in the system can thereby contribute to a restructuring of the entire organization, depending on how the misfit is compensated for through workarounds and bricolages. The bricolages that are digital can spread from user to user rapidly and efficiently in an organization, which could further empower them as their own tools for organizational ordering, perhaps even contributing to a shift in the division of labor. It is already observed that successful bricolages spread inside departments, and that users who move between departments bring along their creations and constructs to employ in their next position – because they consider them as good tools that they know well and feel comfortable with.

Based on the results of the research conducted in this thesis, the types of workarounds and bricolages shown in Table 3.1 have been identified, forming the beginnings of a typology of workarounds. It is the belief that some of these might be identified in other organizations as well, and that thereby, the production and maintenance of a typology of workarounds and bricolages will be beneficial to the field, those who design for it and those who study it. The typology is not constrained to a certain axis or category. Its contents transcend and intertwine itself, and the types are neither inclusive nor exclusive.

The types presented are positioned on an axis ranging from the workaround as the most lightweight and ad-hoc embodiment of the effect to the bricolage as a more complex, heavy-set and long-lasting construct. They are presented in the same order as they have been discussed in chapter 3.2. Each type of workaround or bricolage has been connected with a number of characteristics.

These characteristics, which will be detailed shortly in Table 3.2, represent the traits by which these types have been identified. These characteristics have been discussed in chapter 3.3 and 3.4.

Table 3.1 Types or categories of workarounds and bricolages observed in the study, and their characteristics.

Type of workaround or bricolage	Characteristics	Positioning
Communication outside the formal system	Lack of knowledge Shared misunderstandings Peripherality Transparency	Workaround
Cheating the system	Peripherality Cheating Constructed rigidity	Workaround
Keeping track and archiving	Lack of knowledge Keeping track Missing views	Workaround/ Bricolage
Restructuring and reinterpreting	Lack of knowledge Peripherality Constructed rigidity Pride in the self-made Privacy Transparency Missing views	Bricolage
Bridging to the formal system	Constructed rigidity Pride in the self-made Missing views	Bricolage
Helpful disservices	Lack of knowledge (perceived) Peripherality	Workaround

The first type of workaround identified relates to the **communication that takes place outside of the formal system**. Naturally, some communication needs to take place outside of the formal system – a user that does not have access to the system, but who needs to perform a purchase, must communicate his needs to a purchaser in some way. This is not seen as a workaround in this setting. Rather, the exchange of information where the formal system already provides the ability to do so is seen as a workaround.

In this case in particular, we have seen how the lack of knowledge of the formal system's abilities has contributed to this effect. The workaround can also happen when two parties share a misunderstanding about how the other party prefers to communicate, and the effect is also linked to the prevalent feeling of peripherality to the system.

The second type of workaround relates to how users **deliberately cheat the system** by inputting incorrect data. Gasser (1986) describes a similar situation, where users who were aware of certain shortcomings in the system input false data to get a desired and correct result. The situation observed in this study differs in that the users actually desire an incorrect result, making them appear more compliant to upper management. This type of workaround strongly relates to feelings of peripherality to the system combined with a feeling of constructed rigidity, in the sense that the

users fight not against technical aspects of the system but rather against limits that upper management put in place deliberately.

The third type of workaround discovered is the **creation of alternate views, lists and archives for data**. This encompasses both physical and digital creations. On the lowest level, these workarounds can be caused by a lack of understanding how to access the desired data in the formal system, creating the need to maintain separated lists. This kind of workaround can also occur when the formal system does not provide views of the data in the ways that the users desire.

Fourth, users **restructure and reinterpret the data** on their own when they feel that the formal system is actually a secondary system, and their own construct or bricolage, is the primary system. In these cases, the formal system appears peripheral to the primary work being done by the user, which is much better represented in the tools that the users build themselves. These workarounds can also be seen, to some extent, to counteract the accountability and ordering imposed by the formal system, and as a channel for the users to display their prowess in creating their own tools and interpreting their own work. The category carries a very strong connotation of peripherality, suggesting that many

Fifth, the users also venture beyond their primary work when they engage in creating **tools that bridge the data from the formal system** with their own constructs and interpretations of the data. These tools structure the data into other views that give greater meaning and effect to the users, and overcoming some of the limitations imposed in the formal system in itself. This type of use differs from the previous type in that these workarounds do not signal a peripherality from the system, since they to such a great extent rely on data pulled straight from the system through a tool provided by the vendor. They are, however, an excellent way for the users to show their own skills and abilities and engage in their own textualization of the work processes and organization.

Sixth and finally, it was also found to be common for the two different help desk organizations to perform what can be described as **disservices towards the users** of the ERP system. In these instances, instead of teaching the users how to perform certain actions, the help desks chose to finish the work for them. This contributes to a reduction in the users' belief in their own abilities to perform their primary work, together with feelings of alienation or peripherality from the intents of the system. This situation could be responded to by the users creating their own tools in which they will not be at a loss for what to do.

These types of workarounds are prevalent in this study, and can possibly be recognized in other organizations.

All of these workarounds share a set of characteristics that define their properties and to some extent their causalities. A summary of these characteristics is given in Table 3.2. The

characteristics are ordered beginning with the material and tangible characteristics, such as keeping track, to more abstract and political characteristics, such as those of peripherality and constructed rigidity.

Table 3.2 Identified characteristics of workarounds and bricolages in situated use.

Characteristic	Description
Lack of knowledge	<ul style="list-style-type: none"> The workaround is a result of lack of knowledge or ineptitude at employing the formal system
Keeping track	<ul style="list-style-type: none"> The workaround attempts to facilitate refindability and overview of information or objects that the user has previously worked on The workaround or bricolage is a tool for sharing information between coworkers
Shared misunderstandings	<ul style="list-style-type: none"> The workaround exists because several of the participants share a misunderstanding of how the work should be done The workaround is a result of some long-kept routine that cannot be accounted or reasoned for
Missing or insufficient views or structure	<ul style="list-style-type: none"> The workaround makes up for missing views or insufficient data structures presented by the formal system The workaround restructures the flow or order of work imposed by the formal system The workaround or bricolage is a tool for coordinating activities among coworkers
Detective work	<ul style="list-style-type: none"> Work that does not belong to this user, and necessitates the discovery of novel processes or solutions The user is at a loss for both the “what” the “how”
Privacy	<ul style="list-style-type: none"> The workaround seeks to compensate for the accountability imposed by, or degree in which actions are made public in, the formal system
Cheating	<ul style="list-style-type: none"> The use is an attempt to overcome or surpass some constructed set of rules
Pride in the self-made	<ul style="list-style-type: none"> The workaround or bricolage is an expression of skill at the tools in which it was made, or understanding of the process it interprets
Transparency	<ul style="list-style-type: none"> The workaround or bricolage is invisible to the user as a system or construct on its own The workaround or the formal breaks down and turns opaque
Misfit	<ul style="list-style-type: none"> The formal system is inappropriate or inadequate for the given work
Peripherality	<ul style="list-style-type: none"> The formal system is perceived of as peripheral to the primary work of the user The formal system alienates the user or their department The workaround or bricolage becomes the users primary system
Constructed rigidity	<ul style="list-style-type: none"> The formal system attempts to order the work in a way that the user does not desire The formal system imposes restrictions or requirements that the user does not want to comply with The restrictions of the formal system are perceived of as meaningless or superfluous

Workarounds and bricolages can be measured against these characteristics in order to facilitate their categorization and interpretation. Not all of these characteristics need to be employed in the analysis of a situation in which situated use differs from planned use, as they are applicable in different situations. The description serves as a reference for when the workaround or bricolage can be said to have the individual characteristics, and what the cause or reason for its existence might be.

These characteristics are all products of the grounded theory analysis, and are all present in varying degrees in the situated use studied. Some of the most pronounced characteristics are

identified for each type of workaround given in Table 3.1. The combination of the types and their characteristics serve an important point in informing us that some of the types of situated use that deviates from the plan presented by the formal system are complex and many-layered.

For example, the keeping of secondary lists or archives might be easily explained by how the user is not aware of how to use the proper functions in the formal system. However, the restructuring and reinterpretation of work observed presents a much more complex case, combining peripherality, constructed rigidity and transparency in a web of circumstances. Not every instance is decidedly so complex to analyse, but the effects in their entirety bear connections to all of the given characteristics.

System developers should take such characteristics into account when designing formal systems. In many cases, the workarounds are responses to the desire for flexibility in these characteristics – i.e., the ability to customize views, the ability to find objects easier, customizing what is private or not, overcoming rigid rules imposed by the system and so on. The heeding of these characteristics will not lead to an elimination of all workarounds and bricolages, but will rather ensure the relevance and usability of the formal system in the situated use going on at all levels of the organization.

While the formal system is rigid and scripted, the desires among the end users present in the situated use is the ability to flexibly adopt the use as the needs and purposes vary. This flexibility is present in the workarounds and bricolages, but less so in the formal system.

Star (2010) argues that one should look at the leftover categories of “*not elsewhere categorized*,” “*none of the above*,” or “*not otherwise specified*” (Star 2010, 614) in the development and standardization of a system, because these categories will indicate the actual needs on the end of the user. This is what the typology and characteristics of workarounds represent to the formal system – these are the “*others*” and “*outsiders*” (Star 2010, 615) that the formal system does not accommodate, where the true adaptation of the system is prevalent and visible. Schmidt and Wagner (2004) found that an ordering system, no matter how detailed, will always experience a collection of items that do not fit and must be placed into leftover categories. Through flexible and adaptable ordering and structuring systems, however, these orphaned items can be swept into the ordering structure, thus improving the ordering structure in itself.

It is the support for this very richness and flexibility that should be reflected in these tools, because they are the manifestations of the human side of computer-supported cooperation:

“CSCW encompasses a wide range of control techniques. In many ways this is to be expected; CSCW is essentially about supporting the rich patterns of inter-personal cooperation. This richness should be reflected in the provision of control within CSCW

systems, and the underlying technology should support rather than constrain this process.” (Rodden and Blair 1991, 58)

Both the typology and the characteristics might vary between different organizations or even different user groups within the same organization, and one should therefore be cautious in generalizing these results.

The typology and characteristics further serve to point out that the existence of workarounds should not be conceived of as exclusively a result of the system being “*hard to use*” or the user “*not using it enough*”. Some of the users who spend almost all their time in the formal system, and have done so for years, are very avid users of workarounds and bricolages, like Dina and Fabian. This was illustrated in one of the interviews, querying Eric about WEF, the lightweight web-based employee front-end to the ERP system:

“<How do you perceive registering working hours in WEF, then?>

I do a lot of hours, so for me it’s very simple.

<It’s simple because you do it a lot?>

Yeah.

<Is it simple in general?>

Maybe not. It might be harder to make a travel expense claim if you haven’t done it [before]. I also do a lot of those, so it isn’t a problem for me.

<So to make the travel expense module easier to use people should travel more?>

Eh, no... definitely not. [laughter]” (Eric)

The prevalent view is that the primary obstacle to proper use of the system lies solely in the lack of knowledge in how to use the system properly. The results show that this view is not sufficient to explain all the ways in which situated use takes place, how the system is being worked around and why there exists alternative systems that take over the responsibilities of the formal system. Lack of knowledge is certainly seen, but it becomes only one of twelve different effects that characterize the observed kinds of situated use. It is insufficient, then, to talk of only “*too difficult*” or other similar effects by themselves. Situated use remains multi-layered, and its logic does not always present itself in plain sight.

This chapter has shown some of the instances of actual situated use of the formal system. In understanding and analyzing this use, we have created both a typology of the ways in which situated takes place, and identified a set of characteristics that sets these apart and enables us to work with them as both workarounds and bricolages.

We will now take a look at the results of an opinion poll about the ERP system that was conducted in the organization, and see whether the findings presented in this chapter can be extended and recognized in the organization on a bigger scale.

4. Contrasting with Quantitative Data: An Opinion Poll

Simultaneously with the beginning of the data collection that forms the basis of this thesis, the department responsible for development and implementation of the ERP system decided to conduct an opinion poll amongst all of the end users of the primary computer platform. The purpose of the opinion poll was to assess whether the end users believed that the ERP system would contribute as a means of cost- and resource conservation, planned future system releases included. The poll was distributed to all users who had roles other than the basic employee role in the web-based ERP system front-end (called WEF), and gathered a considerable number of respondents considering the size of the organization ($n = 2.254$). The complete set of questions and response options in the opinion poll is presented in *Attachment 2 – Opinion Poll*.

While a timely coincidence, it is important to note that the execution of the opinion poll is not related to the research presented otherwise in this thesis, and that the decision to release data from the opinion poll for comparison in this study was made after the opinion poll had been completed. As such, the indications gathered through the analysis of the opinion poll data in the context of this study can only be seen as supporting or contradicting and not as affirmative or dissenting. However, comparison of qualitative data collected in an in-depth setting to quantitative data collected from the entire organization appears to be fitting and desirable for such a study.

Of particular interest is the fact that the poll asked whether the respondents “[have] developed alternative ways to work, where one should really use [ERP system]?” (question 7). Undoubtedly designed to create data of compliance or acceptance measurement, in the context of this study, the question is immensely relevant as it in fact queries the users whether or not they employ workarounds. Combined with other questions in the poll, it allows us to group responses by the conscious use of workarounds, hopefully providing valuable insight. Accordingly, the existence of this question compelled the use of the data in this study.

As discussed previously in this thesis, many workarounds become transparent to the users and are thereby not seen as workarounds, and as such it could be expected that the positive responses to question 7 come from those users who employ intentional workarounds that are obvious for the users themselves. It follows then that an unknown number of users might employ some form of workarounds that are transparent to the users and are thereby lost in this poll. Further, it is important to note that this poll question was worded prior to the release of data for this study, and as such must be interpreted as an effect analogous to that of the workaround, but

perhaps not absolutely identical. Regardless, 36 % of the respondents provided a positive response to the workarounds-question, indicating that about 1/3 of the respondents employ what can be roughly interpreted as opaque workarounds.

Users who responded positively to the workarounds question were then asked to explain the cause of such practice, given the choice between four different causes of workarounds (Table 4.1). Further interesting questions were those about the end users perception of the adequacy of their own training as related to the job that they perform (question 9), and also whether they perceived that they had the correct types of access to various operations in the system to get their job done (question 8).

We shall now interpret the responses to the question of workaround cause before we move on to the other questions, some of them grouped by the workaround causation question. Intermittently in the text there will be some discussion relating the results from the quantitative analysis to the other findings presented in this thesis. To facilitate the discussion, we will have to break down the question.

The principal question of the employment of workarounds, then, is broken down as shown in Table 4.1. One must here note that the questions are not designed with this study in mind, and their intent and wording must therefore be considered before the numbers can be interpreted. The first and second cause of workarounds seems highly overlapping, as is also shown in their response rates. There is a nuance to be gathered in the causality embedded in the response; the first response (*"Already had working processes"*) implies a satisfaction or acceptance of the current regime, while the second response (*"Did not wish to change established methods of workings"*) seems to imply more of a resistance towards change or governance, than just the acceptance of the established procedure. It is difficult to judge whether such a distinction was apparent to the respondents, however, and the choices therefore seem fairly overlapping. Regardless, even combined, they represent a minority of the respondents. However, Figure 4.4, we see that there is some pattern emerging in the data when structuring by this response, suggesting that the questions indeed bear differentiated meanings to the respondents, which will be discussed later.

The third and fourth response options are more crystallized in their wording and provide greater distinction that hopefully will lead to greater usefulness. The third option (*"[ERP system] doesn't cover our needs"*) apparently grounds the existence of the workarounds in a perceived inadequacy, gathering up 27 % of the responses. However, it obviously does not inform us if this apparent inadequacy is grounded in a real lack of features in the system or if it caused by insufficient knowledge of the available features of the system. The lack of any other data prevents us from seeing what such "needs" might be, although the data gathered otherwise in this study can suggest

some areas wherein users feel discrepancies in the system. The fourth response alternative ("*[ERP system] has a too high threshold of use/too bad user interface*") might seem like the easiest to understand, while simultaneously being hard to interpret. Formal tools such as ERP systems generally present a user interface that is unlike those that are known from more traditional office support tools, such as Microsoft Office, which might explain why the system is perceived of as too hard to use.

We see, then, that the question and their responses provide some insight; however, the usefulness of such a question might be increased by provisioning it as a multiple choice-question so as not to limit the available options. We shall now continue to analyze some of the other questions in light of these responses.

Table 4.1 Self-reported cause of workarounds among users who reported knowing of workarounds.

Cause of workarounds	Popularity
Already had working processes	7 %
Did not wish to change established methods of working	7 %
[ERP system] doesn't cover our needs	27 %
[ERP system] has a too high threshold of use/too bad user interface	47 %
Other/don't know	12 %
Total	100 %

We will now continue by breaking down some of the other questions by response to the workarounds questions. The first example is a self-estimation of how much time per day is spent employing the ERP system, the responses of which are given in Figure 4.1. This question becomes interesting when looking at some of the internal user roles documentation, in which it is required that a user seeking the role of purchaser must be able to prove that he or she is going to be spending 50 % or more of her workday performing purchasing tasks, wherein the ERP system is the central component. The rationale for this requirement is that user who spends more time using the ERP system will achieve higher proficiency at their tasks, and it therefore makes sense to require that a minimum number of users are spending as much time as possible performing these tasks.

In addition, it could be proposed that users who spend more time employing the ERP system will get better self-insight into what augmentation might be necessary, and would thereby be more likely of employing workarounds. Conversely, it could be argued that users who spend little time employing the ERP system will not achieve the level of proficiency required to accomplish a wide range of tasks in the system, and would therefore be more likely to develop tools for working around what they might perceive as shortcomings of the system. The responses to such a question should therefore be indicative of what scenario is more likely.

In reality, neither theory seems to hold ground. The data shows that both the full-time and low-time users are those who employ workarounds the least. The peak seems to appear in the third quartile, thereby not directly supporting either of these theories.

On the subject of the validity of the requirement to spend more than half of the workday in the ERP system to perform the role of purchaser, we cannot draw any conclusion, either. It is important to remember that these results contain data from other roles beyond that of the purchasers, and the time requirement does not exist for all roles in the system.

In summarizing this question, we see no clear relation between how much the ERP system is employed and the prevalence of workarounds. This suggests that workarounds are as common, or uncommon, for casual users as they are for full-time users of the system. This can be a significant insights, informing us that for example studies of situated use should not only consider full-time users of studies, but also the more casual users who might not even identify themselves with the functions that they are performing as their primary task.

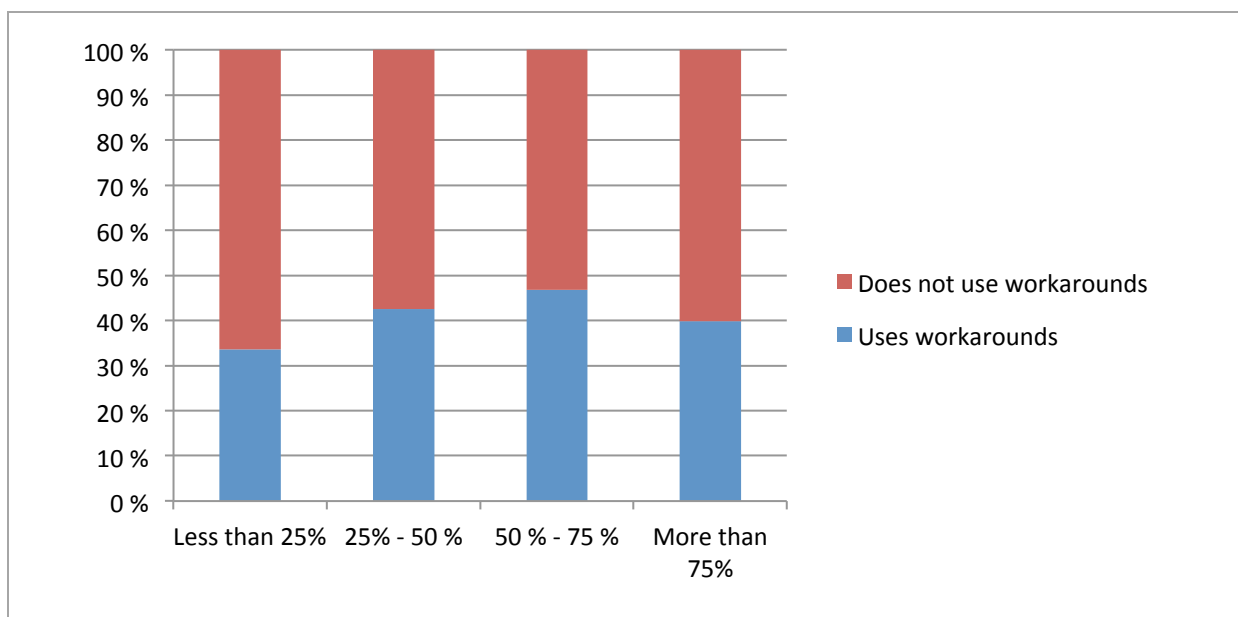


Figure 4.1 Portion of workday reported spent in the ERP system vs whether the users knew of workarounds⁴.

The next question presented for analysis is a self-assessment of access levels in the system (*"I feel that I have the necessary access in [ERP system] to do my job"*). The subjects interviewed for the primary data collection for this thesis often spoke about access and how they felt that they were prevented from performing their tasks because of restrictions on what they were allowed to do in the ERP system. This question is also interesting in the context of workarounds. As has been

⁴ The data behind the graphs in this chapter is given in attachment 3.

discussed, the bricolage is not only the product of a person who seems to have a better vision of how some tasks can be performed, but also the product of a person who fails to accomplish the task in the way she is taught, and therefore devices her own way of solving the problems at hand.

Assuming that some users are prevented from performing their tasks not by lack of skill but by lack of access to the desired features of the ERP system, we can break down the question and see if this might hold true.

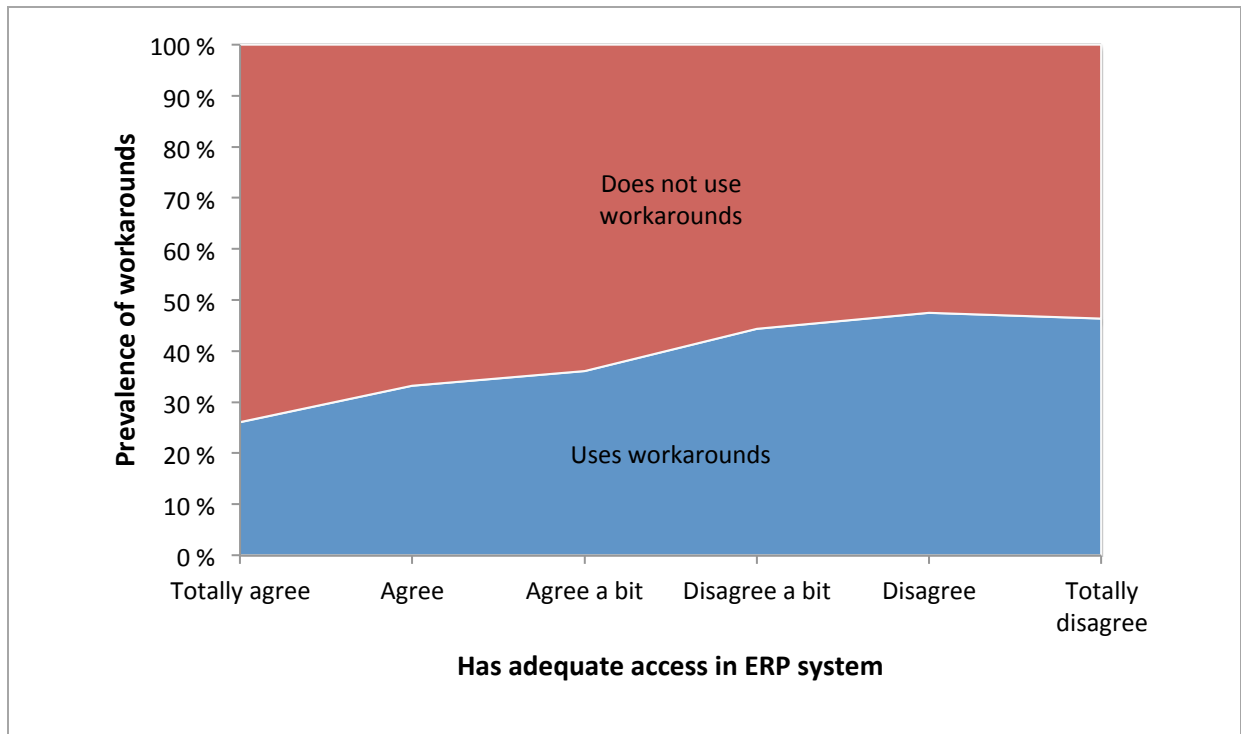


Figure 4.2 Perception of adequate access in ERP system vs prevalence of workarounds.

As the data Figure 4.2 reveals, there seems to be a fairly close and almost linear correlation between the prevalence of workarounds and the perception of adequate access rights in the ERP system – workarounds are almost twice as prevalent amongst the users that feel restricted by the system, at 20 percentage points over the users who feel they have adequate access.

This can suggest that some workarounds are the products of users who simply have become technically restrained from performing their tasks and therefore have chosen to do so outside the ERP system. It can, of course, also be the case that some of these users are simply unaware of how to perform the tasks properly, or which interfaces to use, and therefore suffer from the perception that they cannot access it. Still, the data is so telling that it seems difficult to ignore.

The fact that workarounds are less prevalent amongst users with adequate access rights can also suggest that the need to create workarounds is an artificial one created by the limitations imposed by the system. In other words, it can indicate that some workarounds are not required to

go about the job at hand, and assuming that the bricoleur will readily develop her own tools should she determine that the bricolage would provide a better way to go about the task, we can infer that the ERP system actually is superior to the workarounds that never came into existence. This notion might also be supported by the next question.

The responses, then, seem to suggest that the prevalence of some workarounds increase as access to various components of the ERP system is restricted. Workarounds are still being used by full-time users, but perhaps these workarounds are of a different kind than those that are lost as we move from one end of the scale to the other. Acknowledging that only 55 % of the users either “Agree” or “Totally agree” that they have adequate access to the system, a refinement of what roles are given access to which functions of the ERP system is called for – regardless of workarounds.

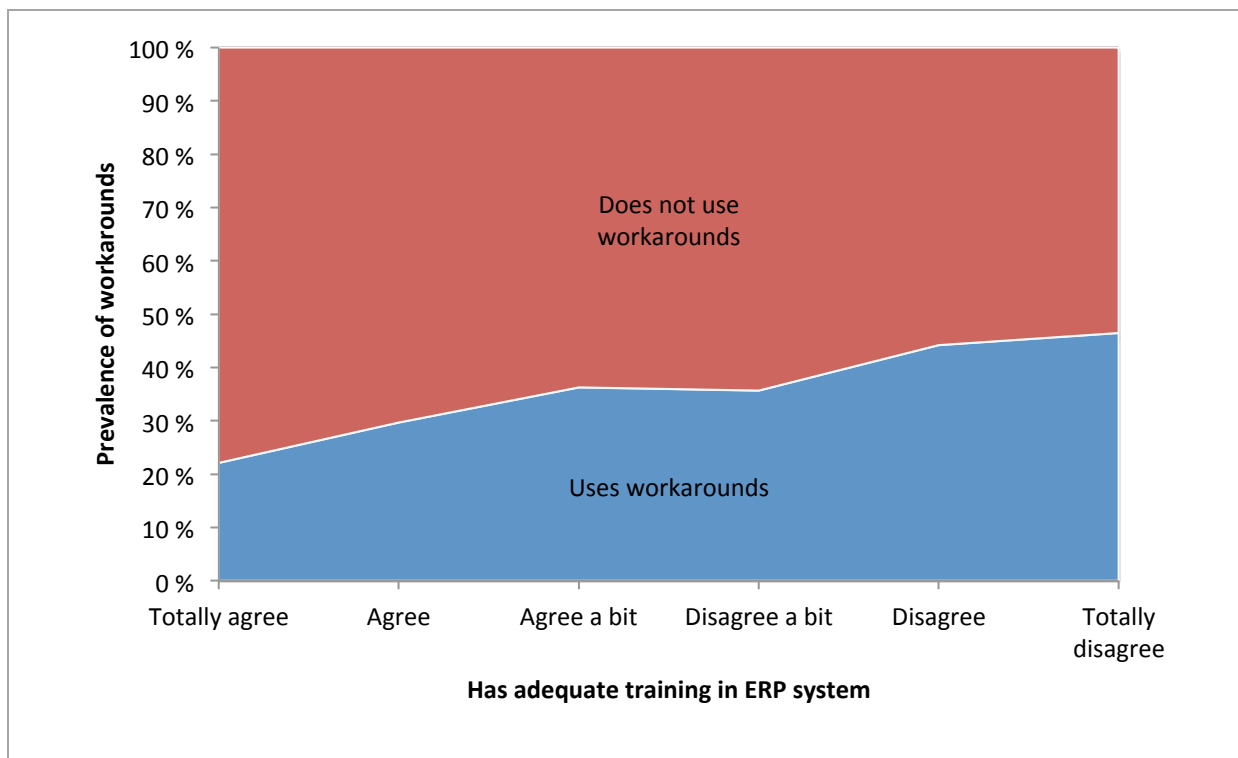


Figure 4.3 Perception of adequate training in ERP system vs prevalence of workarounds.

Turning to the question of proficiency (“*I feel that I have the necessary education in [ERP system] to perform my job*”) in Figure 4.3, we see results not dissimilar to those from the previous question. In this question, the respondents were asked to evaluate their own skills by assessing whether they have been given adequate training in the use of the ERP system. This is, of course, making the assumption that skill comes from training and not from intuition or other immanent sources of wisdom, and that the users responded in this mindset.

Note that the role descriptions of the ERP system requires all users to undertake and pass predefined courses before access is granted, either on-line through a learning portal or as real-world

classroom courses. As discussed previously, the role of purchaser required a one-week purchasing course. This means that all users who are performing these tasks in the ERP system have already undertaken the amount of training judged necessary by the vendor and the department responsible for the development and implementation of the ERP system.

In these responses, we also see a correlation between the prevalence of workarounds and the perception of adequate training. As with the previous question, it is difficult to determine if this relationship is a causal one. On one hand, it can be argued that users who are less skilled at employing the ERP system create their own bricolages that will then serve some of the needs that would otherwise be satisfied by the ERP system. On the other hand, it can be envisioned that the reduced perception of proficiency in the system is caused by the pre-existence of workarounds that prevent the users from achieving a feeling of having adequate abilities in the ERP system.

Again, we can argue that some of the workarounds being created might be created because the users do not possess the necessary skill in the system. This would mean that such workarounds are not inherently beneficial to the work being performed, and the production of these workarounds might constitute a pure duplication of effort. Still, only 32 % of the respondents state that they “Agree” or “Totally agree” to having adequate training, and as such, there seems to be room to achieve a higher skillset amongst the users.

In the next question, we break down the perception of adequate training in the ERP system by causes of workarounds in the hopes of finding some correlation between the two. In this comparison, shown in Figure 4.4, the response column “(No workarounds)” represents the perception of adequate training for those who stated they do not employ workarounds. It can therefore serve as a point of comparison with the other categories.

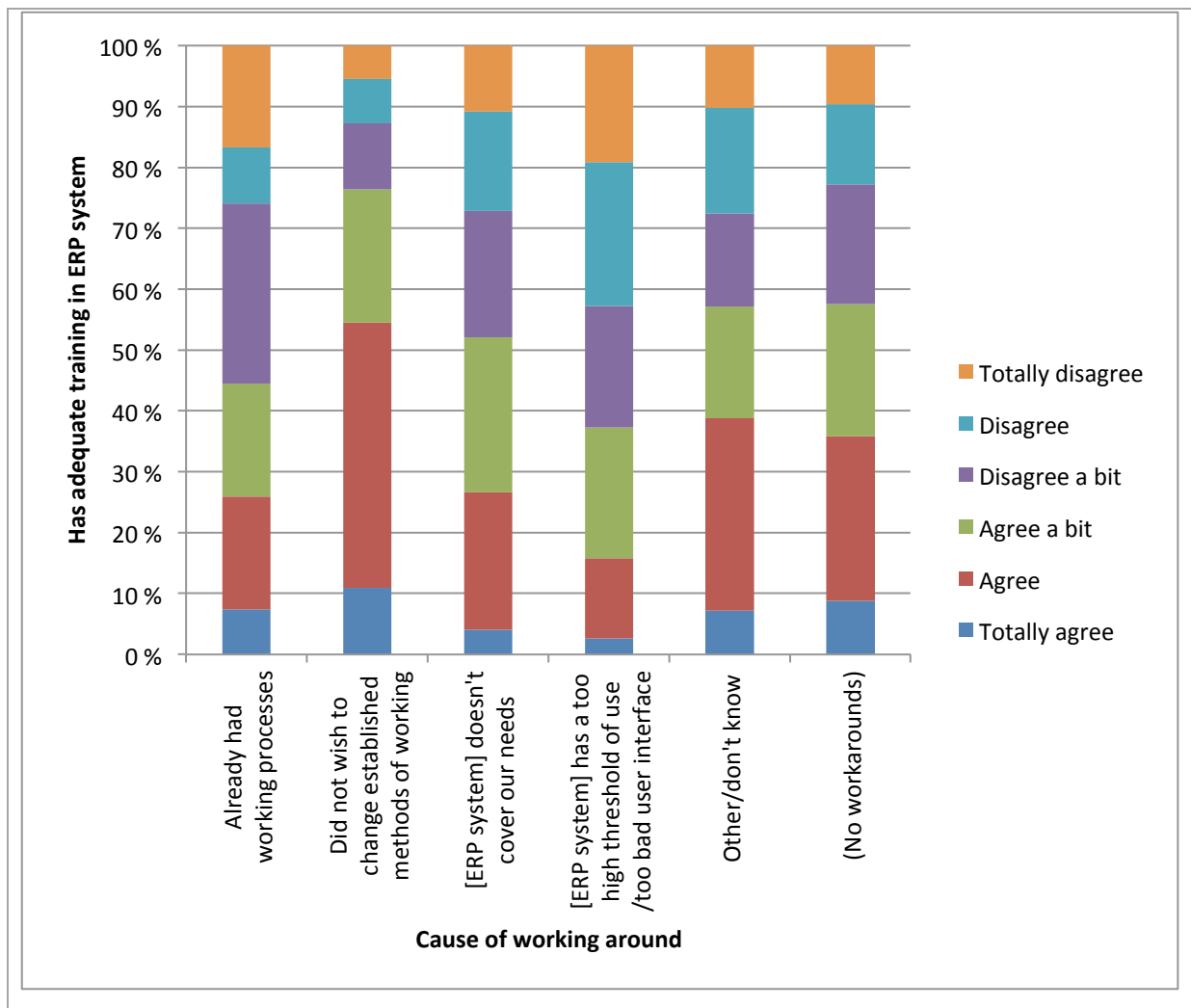


Figure 4.4 Perception of adequate training in ERP system vs cause of workarounds.

Recall that we discussed the similarities of the “Already had working processes” and “Did not wish to change established methods of working” responses earlier in this chapter. As we can see here, although the responses might seem overlapping, there is a clear difference in the respondent’s choices. Note specifically the responses for those who “did not wish to change”. Particularly, the numbers and the heavy weight of those who “Agree” or “Totally agree” that they have adequate training in the ERP system, totaling 55% of the responses. This can suggest that the group of those that “did not wish to change” have a much higher level of self-reported competency in the system versus the other groups, or more significantly, that the users that report to have the highest competency in the system are also those who made a conscious decision to work around it (“wish to change”). This could signify that the workarounds present in this group are created or maintained because the users might honestly believe that they have found a better solution than the ERP system, hinting at a very opaque and deliberate workaround or bricolage.

Further, perhaps a bit unsurprisingly, we find that the group of respondents who say that they employ workarounds because the ERP system has a *“too high threshold of use/too bad user interface”* are also those who report the least adequate training/competency in the system, where 63% of the responses land on the negative side of the scale. In other words, the users that feel that they cannot master the ERP system technically are also those who feel they have inadequate training. This is significant, as we recall from Table 4.1 at 47%, the category encompasses almost half of all the self-reported causes of workarounds. A significant portion of workarounds thereby must stem from the inability to perform these tasks in the formal system.

4.1 Acknowledging the Opaque Workarounds

These responses, then, can suggest that there exists multiple categories of workarounds that surely carry different causalities and attributes. This is significant, since it underscores the need for the understanding of the diversity of workarounds, and further that the organization should not seek to eradicate the existence of all workarounds in themselves, but might rather be better off by offering improved access controls and training, and then studying how the remainder of workarounds are either benefitting or hindering the intentions of the ERP system.

In particular, the results in Figure 4.3 indicate that there might be two major groups of workarounds in existence: One group of workarounds that stem from the inability to perform the action properly in the formal system because of lack of knowledge or training, and another group of users who are skilled at employing the formal system but still choose to employ workarounds because they somehow perceive them as preferable to the formal system.

These results are not unlike the findings indicated earlier in chapter 3.5. In particular, we saw that some workarounds were related to a lack of knowledge of the abilities of the formal system, while others were related to missing features and a desire to create a more suitable data structure in a bricolage outside the formal system.

It is possible that some of the self-reported cases of low training in the ERP system might be brought on by the helpful disservice effect from the help desk discussed earlier. The effect certainly reinforces the disbelief in ones own abilities, which would contribute to a feeling of the formal system being too hard to use. Regardless of the cause of the feeling of insufficient training, the data surely reveals a correlation between the prevalence of workarounds and adequate training levels.

However, some workarounds might also be the result of being restricted by the lack of access to the proper functions of the ERP system. This suggests that not all workarounds provide a superior way of performing the tasks, as it is not possible to say if these tasks would be equally or better performed in the formal system since the users cannot access it.

Workarounds as a result of insufficient access to parts of the formal system did not appear during the analysis in the previous chapter. This could indicate that this problem is not very prevalent in the purchasing regime, suggesting that access is indeed adequate, or that users simply did not speak of it. However, access restrictions can also be seen as a form of constructed rigidity, thus tying the effect into the findings presented earlier.

We have also seen that workarounds are distributed fairly evenly regardless of how much time the users spend in the ERP system, suggesting that both casual and full-time users are all likely to work around the system.

In closing off this discussion, it is further important to note that the data gathered in the opinion poll spans across many different functions and departments of the organization, while the data gathered for other parts of this study is limited to purchasers or financial controllers. To some extent, this indicates that some of the findings presented in the previous chapter could be extended to cover other functions in the entire organization, while also suggesting that some workarounds are profession-specific. For example, workarounds as a result of inadequate access did not appear in the previous analysis.

The analysis of quantitative data provides valuable insight, and has contributed to a justification of some of the types and characteristics presented in the previous chapter. However, the data presented in this chapter is likely limited to opaque workarounds since the very nature of a transparent workaround is that the user is not aware of employing it. Therefore, in amending a qualitative study with quantitative data, this data must be viewed as both supplementary and indicative to the main findings.

5. Summary and Discussion

Studying situated use remains as challenging as ever, and no method can ensure the production of any truth or insight that is universally true. Through pervasive studies we are still able to understand more and more of what is actually going on and how the artefacts and systems we create are used in the wild.

A study of situated use is obviously shaped by how one chooses to interpret the use that is observed, and how one should go about making the distinction between what is and what is not happening the way those who devised it supposed it would it would happen. This study borrows from Suchman (2007), who distinguishes between the different perspectives of *plans* and *situated actions*. The *plans* perspective supposes that the plan is a necessity of action, and that the plan embodies the action by reducing it into steps that can be described as almost commonsensical or inherent to the actor. Thus, the plan prescribes the action that will take place, and is the sole reason why action happens the way it does. *Situated action*, meanwhile, assumes that the plan is a part of the action taking place. The unfolding of such situated action is also guided by the specific circumstances in which the activity unfolds, and interaction around it must encompass “*both a sensitivity to local circumstances and resources for the remedy of troubles in understanding that inevitably arise*” (Suchman 2007, 69). The underlying plan, then, presents the intent of the action that might take place, allowing the actor to decide upon what particular steps to take in any given situation, almost as if picking from a smorgasbord of interpretation – the action is situated and does not take place until enacted *in situ*.

These perspectives are useful in interpreting how situated use can differ from planned use. Interpreting the system and its imposed order of work as a plan of action, it becomes easier to distinguish when this plan is not complied with, resulting in instances of deviations from said plan or in breakdowns. Similarly, in interpreting situated action, we can acknowledge how the system becomes not a scripted plan but a resource for the user in her desire to accomplish a given task, or how the actions performed in the situated use show that the system is insufficient or otherwise ill-suited, perhaps because another set of tools will be employed in its place. These are exactly the perspectives that have been taken in this study.

From the *plans* perspective, we see that the formal system does offer an expectation about how the work should unfold which is embedded into the mechanisms offered by the system. The system does not separate functionality from content, as each piece of data appears as a work task, and all functionality is made available through a transaction-oriented approach.

The users, who are the ultimate enactors of the *situated action*, are less interested in this approach. They prefer tools that do not impose any particular order on the work, but which support a broad range of actions on almost any object at any point in the process, where they can supplement the data objects with relevant pieces of information that aren't bound to strict rules of validation or coherence. An inherent conflict arises, which is ultimately answered by the users creating their own tools, in preference over the order imposed by the formal system.

The system, then, attempts to order the work so that it can almost flow automatically. The users, on the other hand, desire access to and overview over a breadth of information objects that they can act upon as need be. This contrast is also apparent under another name – *automating* versus *informating*.

Zuboff (1985) argues that the introduction of technology into a given field primarily takes one of these two paths. When automating, "*the aim is to replace human effort and skill with a technology that enables the same processes to be performed at less cost and with more control and continuity*" (Zuboff 1985, 8), whereas when informating, "*technology can be used to create information*" (ibid.), which is then provided to the user broadly and without discrimination, whereupon the user can choose what information objects to work with and how. The development and introduction of modern IT system can be seen as either a desire for automation and control through automation, or a path to upskilling, democratization and innovation through informating. We can see the ERP system as an attempt at automation and control by upper management, while the response from the lower levels of the organization is one of informating and innovation. Zuboff argues that it is only through informating that the true power of technology can be leveraged; not as a tool for imposing ordering and accountability, but as a provider, processor and communicator of information.

As is apparent from the interview with Eric, the organization does attempt to utilize the possibilities of peripheral data production through the process of informating (Zuboff 1988) – for example in how information from the purchase order processes can be used in other processes such as inventory control, warehouse management and budgeting, by automatically triggering specific sub-processes that can be hidden from the end users. For example, the purchase of an item that has to be tracked in an inventory list would automatically update the inventory. However, the other interview subjects seem fairly oblivious to this: Fabian exclaims that he "*doesn't know what's going on behind-the-scenes in [the ERP system]*" and shows no relation to the other processes that might have desires to use the data he inputs, thereby running the risk that the data he provides is unsuited for informating and ruining the potential for creation of multiple purposes for the same data.

The fact that the users are not aware of what is going on behind-the-scenes in the formal system not only prevents them from adapting the data they input to the system, but it makes the users unaware of what the possible consequences of the upkeep of a workaround or bricolage might be. If a department decides to employ the ERP system at its bare minimum, unaware of what the data given to the system might be used for, then the other processes and functions that depend on these data are likely to fail. For example, the should upper management or the accounting department decide to extract some statistics from the system, they might find that the data is insufficient because the users keep most of their records in an alternate system.

This is not an argument to ensure the use of the formal system by force, but rather a call for the formal system to adapt the features and benefits of workarounds and bricolages that make them so attractive, combined with educating the users on how data in the system is beneficial for other parties. The latter has though proven to be a repeating difficulty in CSCW – users are seldom enthusiastic about doing work that is not for their own direct benefit. Grudin names this one of his challenges, calling it the *“disparity in work and benefit”* (Grudin 1994, 97). He argues that it can be countered either by repeatedly demonstrating the benefits of providing such information, or by making it easier to do so. Undoubtedly, should users move away from the bricolages and into the formal systems, this might be easier to achieve.

We can also observe how the formal system presents itself as a method of accountability and ordering in how it imposes sets of rules on how work must take place, and how many actions and data points becomes visible to other users and management. Bowers, Button, and Sharrock (1995) argue that a CSCW system can be considered a technology for organizational ordering and accountability, by which they mean the way in which management utilizes the CSCW system to gain an omnipotence and omnipresence through imposed process and structure, and inherent reporting. The organization is spread across the country with a plethora of units located on different sites where the upper management has little or no chance to keep up a continuous physical presence, and the formal system can thereby be seen as one of the ways in which upper management can exercise its control here.

This is not to claim that the loss of control is regained through the ERP systems by artificially constructed rigidities, but such control is certainly asserted and confirmed through the system. The attempt at asserting control through the system is then challenged through the development of local bricolages, constructs that to some users are more real or true than the formal system itself claims to be.

The process of taking data both from the system and the environment and combining it in systems that are developed, adjusted, tinkered with and evolved continuously is seen in the process

revolving around the systems designed to augment or work around the perceived shortcomings of the ERP systems. Not only the act of the process of bricolage in itself is interesting, but also the intention that the combination of these will create something more than just the sum of the parts. The bricolage process aims to create something more than itself; it will augment the data in order to create a system that can twist and turn the original information into something that gives greater meaning to the user, not unlike informing which seeks not only to computerize, but to further empower both users and process in the process.

The bricolage being equally important as the formal system is particularly true in the instances where the bricolages become the systems that support the work to a better extent than the formal system does – examples of these are Ben's list of purchases that is used to coordinate between shifts and Fabian's construct that not only talks of purchases as abstract objects, but as tangible substances that empower him and his organization, like machines, fuel, food, spare parts and so on. Fabian and Ben have actually created their own tools to facilitate the creation of awareness, as proposed by Schmidt (2002) in one of the incarnations of the term. He categorizes this as a subdivision of the articulation work that a CSCW system supports when one is *"taking heed of unfolding events and of possibly unfolding events; of things being done, of things done, and of things in need of being done"* (Schmidt 2002, 290). The creations fit to this definition, which they support, while the formal system does not. It seems that the formal systems desire to impose a certain order of work has necessitated the neglect of such functionality. But when Ben and Fabian need to coordinated their activities with their co- and shift workers, they create the tools they need to satisfy this demand. These shared bricolages become subject to improvement and adjustments from all of the users of the tools.

And the desire to make and tinker with the bricolage is present even when there is yet to be a bricolage: Adam clearly explains that they *"don't employ any extra lists or alternative systems [...] because we lack the fundamental computing skills"* – clearly implying the desire to order their information to move beyond the inferiority of the present paper-based archive system, a feeling mirrored by Fabian when he talks of how they have tried to augment the practices of the ERP system with barcode scanners and label printers, yet to no success. Adam is subject to the restrictions of the formal system, but he does not have the skills to break away from them by creating his own data structure. This does not subdue the desire to break away from the rigidity and lack of views of the formal system. Since the formal system is peripheral to him and his organization, he desires something else that can become an integral part of his primary work.

Further, consider the accounts presented by Star (2010) of how detail and resolution is lost in the objects that transform information from raw information to formalized and structured data.

Star presents how formalized data structures can fail to communicate nuances of importance: The most touching might be that of the sick child's parents who were asked to keep a log of the child's activities throughout the progress of the disease; the parents diligently filling in structured data in the columns while scribbling their own personal remarks and interpretations in the columns, seemingly in contradiction with the intents of the form (Star 2010, 607) – or the case of the biology researchers' calm and structured paper, built on torn and rustled field notes spattered with blood, telling the stories of a great chase across the laboratory (Star 2010, 606). Such structures negate the human and social, preferring the rigid and analysable. The ERP system works in similar ways – it formalizes, structures and rejects – the granularity of the original information that was communicated to and by the purchaser is lost here, but present in the bricolages. The locally developed systems can be a result of the desire to track these factors.

The combination of loss of what the users perceive as important information in and control of the system can spur the workarounds that are observed – the users' response to their perception of the constructed rigidity and lack of proper views. As Rodden and Blair (1991) point out, the control over the information and its flow exercised by the system is a major problem in CSCW – challenging the dynamics of the human work environment:

“The problem [...] is that presumed control decisions are embedded into the system and hence cannot be avoided or tailored for specific classes of application. This is the root of the problem in supporting CSCW.” (Rodden and Blair 1991, 59)

Rodden and Blair thereby claim that such tools should not embed any control decisions, but should rather separate control or flow of work from the content of work. These actions *are* separated in the constructs created by the users in this study, in the sense that they carry few if any control decisions at all.

Fabian interestingly remarks that he has no idea how other departments have adopted the purchase process to their organization, or how it has been implemented in other places. This suggests that there is little or no exchange of experience or lived practice between the individual users of the application. Yet most departments in the organization seek to develop their own solutions that work side-by-side with the formal system, and it is reasonably safe to assume that many departments must have developed similar structures.

Maybe some planned use practices have gotten lost between the developers and the end users. Some knowledge and experience exchange could be profitable here; Fabian remarks that there *“is no environment for two-way dialog in the organization,”* commenting on what he perceives as a non-inclusive and unidirectional system development process, and the absence of opportunity to give feedback.

The established processes, then, are not followed letter by letter in the real-world settings, but are rather adapted and appropriated depending on the needs of the individual workers. This in itself is not a revolutionary insight; it is a common trait observed in the fields of groupware and CSCW. Ciborra (1997) aptly comments:

“Looking more closely at the stories of how such systems were implemented, and putting into brackets the disconcerting multiplicity of intervening variables in each case, the common trait that characterizes the human organization involved is the great amount of care the members, each in their own roles (managers, designers and users), have spent to incorporate the new technology into their daily work life.”
(Ciborra 1997, 6)

Interestingly, though, a fair amount of the literature examined assumes that it is the organization, its users or the division of labour that will or has adapted as a result of the introduction of new technologies. Less prevalent are critical views on how the software can adapt, or how software can be developed to support adaptability and configurability, and lastly how organizations and technology can adapt together into a state that is beneficial for all.

Given the view that some workarounds might be a natural part of any form of work, it becomes almost impossible to clearly define which conditions that must be met before one can claim that a workaround has come into existence. This calls for a shift from talking about workarounds as a separate kind of artefact into distinguishing different forms of situated work and its intention. Workarounds in themselves should not be seen as unfavourable, but rather as interesting opportunities for innovation, adaptation, development and upskilling of the users. Groupware could subject itself to bricolage and textualization on the part of the end users, and providing the required flexibility for this will be among the chief challenges for integrators and organizations. Many organizations might not be ready for this regime.

For the fisherman, the fishing rod and net are his essential and implicit tools of the trade, and the boat is his vessel for getting there; for the blacksmith, the hearth and the hammer, and for the IT specialist, the computer. The economist or the purchaser, on the other hand, has few physical tools that are necessary for the successful performance of their trade. To some extent, they can be knowledge workers, depending on the capabilities of the mind in their work (Kleinman and Vallas 2001). But they are still subjected to the formal system and its limitations. A worker in a tool-less profession is thereby assigned a tool that imposes a certain order on their work. As such knowledge workers subjected to the formal system, the creation of workarounds and bricolages can be a move towards a restructuring and perhaps democratization of this situation.

Zuboff (1988) points out that workers in general harbor a much greater insight and a much higher skillset at specific tasks than what is required to perform the task itself directly. For example, a factory worker not only knows how to perform his own tasks the proper way, but he also knows what the others in the factory are doing and how his work both builds on and feeds into others work. As such, knowledge of the outlying processes and intersecting task chains in the production lattice extends beyond the mere situated work situation. This notion is somewhat missing in this case. The users of the ERP system are not aware of what the processes are doing behind-the-scenes, and every single requirement and responsibility they have to the ERP system therefore goes unjustified: They don't know why they have to do a certain task in a specific way, or who else might benefit from the information they provide. Such connections are, however, blindingly clear in the workarounds and bricolages that are created. In these creations, the users are well aware of how the data might be reused and built upon by others, which in turn enables them to optimize their own procedures to benefit the entire process.

Does this knowledge of outlying processes make the user a better developer than any other specialist? To some extent, this view legitimizes workarounds and the bricolages as a way of allowing the situated user to extend skill and cooperation beyond their prescribed tasks, regardless of the limitations imposed in the formal system. To Zuboff (1988), this is a key part of informing: It is not just about moving otherwise analogue processes into the digital domain for the purpose of being digital in themselves, but to be able to recognize how these processes and process variables can be utilized in other settings to achieve even greater value than the pure digitization. In this case, the end user, empowered by the differences between what is offered by the formal system and their own needs, does some of this work.

Barley and Orr (1997) tell of a work situation where engineers and technicians are specialist roles filled by persons who are not separated from the other users in the workplace by organizational means, but rather appropriated into their own fraction as a result of the perceived effect of their particular skillset and knowledge. Recalling their description of the emerging engineer-technician, as discussed in chapter 1.1, it might be possible that through upskilling and the provisioning of adaptable tools, this role is not to be taken on as discreet jobs of new workers, but can rather become an extension of everyday work for each skilled knowledge worker. Some users then choose to become their own engineers in creating the bricolages. But this is not for all, as said by Graham:

“Not everybody was born and raised with the computer, there are many who got to know it after it was introduced in [the organization], and are used to maybe not seeing

the possibilities. Plus, it can't do everything; it only gives back what you put in. Nothing happens automatically."

While the provision of flexibility and adaptability might be a key goal for CSCW developers, so will be the optimization of the default states of the formal system. Users like Graham will probably not engage in a lot of configuration and appropriation work, but are rather prepared to be content with how the formal system ships out of the box; for him, this might be more than enough of a challenge. This further underscores the importance of learning from situated use; for users who are not going to change the default configuration of the system, or provide their own bricolages around it, the system needs to be functional with not just a minimal set of features, but with a useful set of features and options as it comes when it is turned on the first time.

*

With the intent of identifying characteristics of situated work and real-world usage of a computer system, the study is empowered by only interviewing end users of the system, save for one who is both a user and a teacher. This has also affected the perspective of the study, as there might be propositions or claims made herein that will go unanswered by top management, but the daily life of the end user remains the same regardless. Some users were sharing the tasks in a group while others took on singular jobs in a more team-like cooperation with the employees who were in need of purchasing and the accounting departments.

We have seen that an exploration of the space between scripted action and actual lived behavior will yield interesting and useful results than can hopefully prove beneficial to the field. Exploration of the difference, or rather the relation, between the map of the formal system and the territory of actual situated use shows us that while not all use is to the letter of the law, most of it remains true to the spirit of it.

Some workarounds are clearly identifiable, when the users act despite the intent of the system, while others should not so easily be seen as dissenting with the spirit of the system, but rather as natural ways to bridge the provisions the formal system and the actual needs of the real world. This work becomes more of a part of the primary work and less its own supplementation to the work that is supposed to take place.

Situated use does differ from planned use in various manifestations across different settings in the organization. We have seen how communication takes place outside of the formal system, for example through what is termed a shared misunderstanding. Further, users are cheating the system when they purposefully falsify information to seem more compliant towards upper management.

Users are also adept at creating their own archives and data structures. Some of these function as tools for keeping track and looking up, while others take on a much more serious role in that they actually structure the data and the work around the data, carrying both context and content. Some of these tools can come about when users feel that they are not the intended audience of the system, through an effect called peripherality. Others create alternate systems because they lack the desired views and data structures in the formal system itself. Common to these tools is that they take over the role of primary tool from the formal system itself, resulting in a further increase in peripherality to the formal system. Some workarounds can be seen as a response to the rigidity of the formal system, a rigidity that is seemingly constructed for purposes that the users do not identify with.

Additionally, the actual or perceived lack of proficiency can be the cause of the existence of some of the workarounds. Some users store and structure data in a certain way because they simply don't know how to do it in the formal system, while other users are under the impression that they cannot do it themselves and must continuously request the assistance of the help desk to solve their daily tasks. They both desire and attempt to provide feedback to the development organization, but such feedback goes unheeded. This might contribute to the upkeep of some workarounds as the desired functionality is not implemented, while also ensuring a peripheralization of the users when they never get their voice heard.

We have also seen how qualitative data, which forms the primary source of data in this study, fairly successfully can be compared and contrasted with quantitative data, in this case data collected from an opinion poll. This can shed more light on a range of issues either by affirming, rejecting or nuancing already gathered findings. As argued by Crang and Cook (2007), such triangulation approaches can provide stories that both converge or diverge, and either outcome might be equally desirable. They do argue that regardless of how many different approaches one takes, no amount of additional methods will be able to produce an exhaustive result. Still, successful combination of qualitative and quantitative data remains challenging in that the results must be treated with scrutiny and care, as always.

The workarounds identified are all examples of how situated use differs from planned use of the system. Some of these workarounds are arguably the natural extension of work required to make work happen. In surveying these ways of working with the system, we can see the need for a language of understanding and describing situated use and workarounds, and the basis of this has been provided. While some workarounds are clearly not necessary for work to happen, others perform crucial tasks and the successful use of the formal system cannot take place without them. Additionally, the creation of workarounds and bricolages can do more than just make work happen –

they can help make work happen better, as they represent the voice and needs of the situated user, who quite possibly possess the best knowledge of the work that is to be performed.

Recall from the beginning of this chapter, that we highlighted that it is the opposing forces of the formal system and the needs and desires requested by the users that in itself becomes the driving force behind the creation of novel forms of and tools for situated action. Berg provides a similar claim, in that "*the generative power of formal tools, then, lies in the very existence of the gap between the workpractice and its formal representation*" (Berg 1997, 153). According to Berg, it is the very existence of this gap that drives work and provisions the tools as able to provide added value: The fact that the system carries regulations and restrictions on how work should be done and how data should be shaped, is the enabler that when combined with the desire for a certain type of action on the part of the user, will lead to added value in the combination of user and system. In Bergs view, neither the user or the system are by themselves enough for work to happen – the worker is positioned as a form of mediator, and together work will be produced.

This generative effect is certainly at play in this case, but more so in that it also gives rise to the development of tools and forms of work that were not there before. The gap between the plan provided by the system and the situated action enacted by the user not only generates work, then, but also generates reinterpretations of this work, as the formal system is not entirely aligned with the interests of the user. These reinterpretations seemingly provide much greater value to the users, and should become excellent reference points for the developers of the formal system.

This is not to paint a universally negative picture of the ERP system. There exists a ubiquitous acceptance with and understanding of the need for the ERP system, and it is in no way flat-out rejected by the departments in the organization. The users are generally optimistic about the system and regard its features as opportunities for innovation and creativity, and the outlook on the future is positive: The prevalent view is one of cautious buoyancy. Not all are universally optimistic, though, and closing the interview, Dina adds contently, "*I have reconciled myself with how it is.*"

Jonathan Grudin (1994) posed eight challenges that he claimed are the primary problems in the field of CSCW. His sixth challenge states that "*the almost insurmountable obstacles to meaningful, generalizable analysis and evaluation of groupware prevent us from learning from experience*" (Grudin 1994, 97). The development of a vocabulary of situated use then becomes crucial to the understanding of and the development for CSCW. The concepts brought forward in this thesis, which are meaningful in our desire to understand situated use of such technologies, is a contribution to this challenge.

6. Conclusion

The research question posed in the beginning of this thesis is two-fold. First, it calls for the exploration of how situated use (Suchman 2007) takes place, and how it differs from the planned use of a system. Second, the question asks how we can interpret and talk about these ways of using the system.

To identify how situated use actually takes place, we have to realize that situated use must differ from planned use, as use strictly according to plan would be unlikely to accomplish any work when it meets the real world: The human is indeed necessary to interpret and adapt the planned use to the actual work situation. Interpreting these differences as workarounds, the workarounds become integral to situated use, and cannot be inherently interpreted as undesirable. Without the workarounds, work would be severely restricted. The ways in which situated use takes place, differing from planned use in the forms of workarounds and bricolages, are shown in Table 6.1.

Table 6.1 Types or categories of workarounds and bricolages observed in the study, and their characteristics. Also shown in Table 3.1.

Type of workaround or bricolage	Characteristics	Positioning
Communication outside the formal system	Lack of knowledge Shared misunderstandings Peripherality Transparency	Workaround
Cheating the system	Peripherality Cheating Constructed rigidity	Workaround
Keeping track and archiving	Lack of knowledge Keeping track Missing views	Workaround/ Bricolage
Restructuring and reinterpreting	Lack of knowledge Peripherality Constructed rigidity Pride in the self-made Privacy Transparency Missing views	Bricolage
Bridging to the formal system	Constructed rigidity Pride in the self-made Missing views	Bricolage
Helpful disservices	Lack of knowledge (perceived) Peripherality	Workaround

While workarounds are small, lightweight and often ad-hoc ways to appropriate the planned use onto the needs of everyday, they are insufficient to describe all the ways in which situated use differs from planned use. Bricolages, on the other hand, are more complex and longer lasting constructs that to a better extent show how users employ would prefer to go about their work.

Conclusion

Bricolages can be the extension of many combined workarounds, but they might also become so significant to the user that they become a system in their own right, side-by-side with the formal system, challenging it as an authoritative source of data.

This is the essential difference that necessitates the distinction between workarounds and bricolages: Workarounds cannot become a system in their own right, as they are totally dependent on the formal system for their existence. Bricolages can, as they represent more than just “*working around*” in that they are a reinterpretation of the structure and order of work. Yet the bricolages are still constricted by whatever means and knowledge the situated user has at any given point in time. This explains why some departments are able to construct and maintain their own bricolages, while others remain working around the formal system.

The types of situated use from Table 6.1 can be positioned as either workarounds or bricolages, as shown in Figure 6.1. In doing so, it becomes clear that the former seems more beneficial to the work than the latter. An important distinction becomes apparent – the use that is positioned as workarounds is less apt at supporting the primary work of the user than the bricolages, which both support and enhance the primary work. The workarounds that have been discovered are generally not aligned with the intent of the primary work of the user, and are perceived as compensations for shortcomings in the system, while the bricolages that have been created have come about to enhance the primary work of the user.

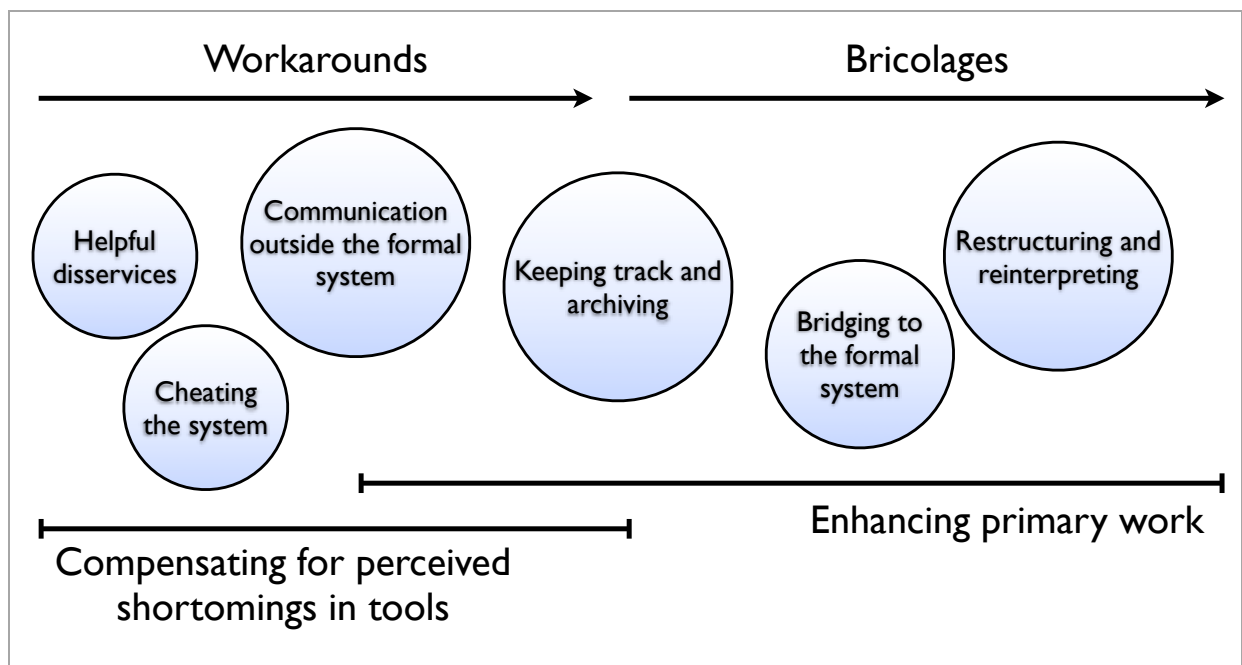


Figure 6.1 Positioning the situated use as workarounds and bricolages reveals that the latter is more beneficial than the former. Size of the bubbles roughly represent their regularity in the study.

To the users, the object of work and the structure of work are more intertwined than the formal system desires. The system wishes to abstract away all the details that is unessential to it, and this renders it inefficient and undesirable as a tool to support the situated work of the user – the user desires a tool that supports her own work and the at-hand tasks.

Further, the system offers little in the way of customization or adaptation, and the users are compelled to create their own tools for supporting work and cooperation, in which they put down their own interpretation of what work should be and how it should unfold. Knowing this, we can seek to design the system to alleviate the burdens that causes the workarounds while learning from the interpretations that the users have put down in the bricolages.

The system should not be adapted to support the workarounds, but it could be adapted to support the bricolages. In particular, in the cases where the user feels that the bricolage supports the primary work better than the formal system, the formal system should be extended to either incorporate the features present in the bricolage, rendering the bricolage redundant, or it should be adapted so that it will interface better with the bricolage on the users end.

The workarounds are not directly undesirable in themselves, as they represent important indicators of how the system is appropriated in situated use. Some workarounds represent hindrances to the undertaking of primary work for the users, and represent excellent opportunities for learning for developers and upper management. Other workarounds are desired, as they allow for the uninterrupted unfolding of work in the situated actions. Further, some situated use can take place through bricolages, which take on a bigger responsibility as they both supplement and challenge the formal system. Representing the users actual desire for the unfolding of work, these bricolages form excellent opportunities for bottom-up innovation and reinterpretation of work. We also see that workarounds that are beneficial and successful can grow to become bricolages.

These ways of differing from planned use can be described and understood through of a set of characteristics, described in Table 6.2. Some of the characteristics can be causal, while others are attributes. Depending on the size and shape of the workaround or bricolage, many or few of the characteristics can be relevant. With these characteristics, one can describe the ways in which different kinds of situated use takes place, and how such use actually has value to the users.

Perhaps the most pronounced characteristics are those of peripherality and constructed rigidity. Peripherality is present in most of the types of situated use discovered, and represents the perceived distance between the intentions of the formal system and the desires of the user. Further, peripherality is both a cause of the production of workarounds and bricolages, and a reinforcing effect that amplifies itself as the user becomes more dependent on the bricolage and less attached to the formal system. Its apparent effect is magnified by the perception of constructed rigidity,

Conclusion

which is the perception that the formal system imposes a certain set of rules or requirements that are not founded in anything but the restrictions themselves. The disregarding of these rules in the workarounds and bricolages leads to further peripheralization, which in turn reinforces the feeling of constructed rigidity and so on, as illustrated in Figure 6.2. Further, the continued relocation of work to a system outside of the formal system will render the actions that the user must undertake in the formal system increasingly opaque, while the workaround or bricolage becomes increasingly transparent. This effect cannot be remedied by simply forcing users into the formal system; the formal system needs to demonstrate relevance and flexibility to attract the users back.

Table 6.2 *Identified characteristics of workarounds and bricolages in situated use. Also shown in Table 3.2.*

Characteristic	Description
Lack of knowledge	<ul style="list-style-type: none"> The workaround is a result of lack of knowledge or ineptitude at employing the formal system
Keeping track	<ul style="list-style-type: none"> The workaround attempts to facilitate refindability and overview of information or objects that the user has previously worked on The workaround or bricolage is a tool for sharing information between coworkers
Shared misunderstandings	<ul style="list-style-type: none"> The workaround exists because several of the participants share a misunderstanding of how the work should be done The workaround is a result of some long-kept routine that cannot be accounted or reasoned for
Missing or insufficient views or structure	<ul style="list-style-type: none"> The workaround makes up for missing views or insufficient data structures presented by the formal system The workaround restructures the flow or order of work imposed by the formal system The workaround or bricolage is a tool for coordinating activities among coworkers
Detective work	<ul style="list-style-type: none"> Work that does not belong to this user, and necessitates the discovery of novel processes or solutions The user is at a loss for both the “what” the “how”
Privacy	<ul style="list-style-type: none"> The workaround seeks to compensate for the accountability imposed by, or degree in which actions are made public in, the formal system
Cheating	<ul style="list-style-type: none"> The use is an attempt to overcome or surpass some constructed set of rules
Pride in the self-made	<ul style="list-style-type: none"> The workaround or bricolage is an expression of skill at the tools in which it was made, or understanding of the process it interprets
Transparency	<ul style="list-style-type: none"> The workaround or bricolage is invisible to the user as a system or construct on its own The workaround or the formal breaks down and turns opaque
Misfit	<ul style="list-style-type: none"> The formal system is inappropriate or inadequate for the given work
Peripherality	<ul style="list-style-type: none"> The formal system is perceived of as peripheral to the primary work of the user The formal system alienates the user or their department The workaround or bricolage becomes the users primary system
Constructed rigidity	<ul style="list-style-type: none"> The formal system attempts to order the work in a way that the user does not desire The formal system imposes restrictions or requirements that the user does not want to comply with The restrictions of the formal system are perceived of as meaningless or superfluous as they are not grounded in real-world requirements or laws

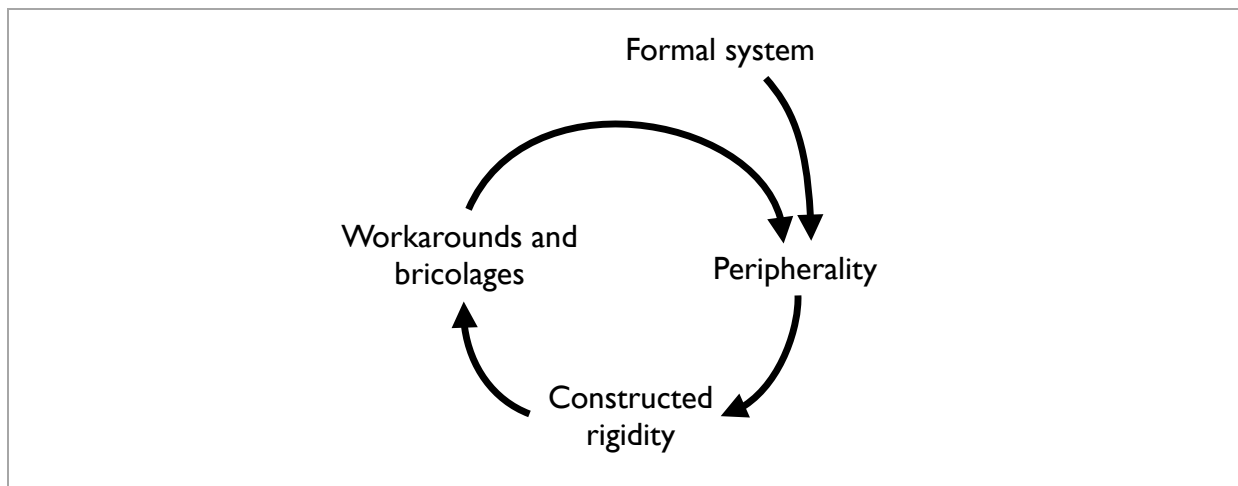


Figure 6.2 The reinforcing feedback loop of peripherality.

The automation of knowledge work represents not only a poor attempt at effectivization, but will in essence demote the work, as the activity is removed from the realm of the knowledge worker – the mind – and into the computer, where the worker is no longer in control of it. When informing is applied to knowledge work, it will be taken a step further, since knowledge work in essence is already about informing: We must provide the users with tools with which to build tools. This is how intelligent technology can provide added value to knowledge work.

“When intelligent technology creates (or provides new access to) information, and when that information is made available to those at the point of production, the essential logic of Taylorism is shattered. For the first time, technology returns to the worker what it once took away, but with a great deal more as well.” (Zuboff 1985, 15)

Zuboff points to the ability of the worker to re-apply her skills in light of this new information, a variation of upskilling that arises at the intersection of new perspectives and new information. It is this extension that we now seek; we have *upskilled* the worker by giving her access and ability to see the details of the work, now let us *uptool* her by giving her the ability to change, adapt and create the way she wants.

Gasser's (1986) notion of misfit as the cause of fitting, augmenting and working around computing is not sufficient as it implies an optimal and desirable state of fit, where in reality, any system needs continuous local adaptation and to some extent innovation. We must therefore understand the situated use in a broader context, such as through both workarounds and bricolages.

In understanding these workarounds and bricolages, we need a language to describe what we see and how these constructs are positioned. The characteristics of workarounds presented in this thesis forms a vocabulary of variables that will help researchers interpret, evaluate and position

these constructs. In doing so, we can design systems that adequately support a range of practices of situated use.

We should seek to design formal systems to better appreciate how situated use actually takes place and how users both need and desire to adapt the work to the situation at hand. Looking at how work is adapted through workarounds and bricolages benefits this. After all, it is only in the situated use that work really gets done – so why should we not learn from it?

6.1 Further Research

We are fortunate enough to live in a time where the ever-increasing demands on computing processing power and transmission bandwidth, provided by the introduction of ever-amassing multimedia platforms, have led us to a situation where we as CSCW scholars can finally study applications with comparably low-bandwidth and low-processing-power requirements like ERP systems without the need to worry about basic technical limitations. The remaining challenges of CSCW, then, become ever more pure as they are less about technology and more about the understanding and support of cooperative work arrangements.

The research uncovers interesting differences between formalized process, or plans, and actual, lived practice and their situated actions. Further research might explore this gap more thoroughly, investigating other methods or artefacts that have been developed and how they are used. Studies of these practices or artefacts might uncover universal approaches that can be implemented into the ERP system or generalized and given as recommendations and templates for the entire organization. On the other hand, further research might uncover that such implementations are not universal, and that these kinds of adaption processes are required in using a formal system.

As we have seen, the subjects of the interviews expressed feelings of being peripheral to the formal system. Does a solution necessitate local adaption and innovation, or should every department be able to conform to the norm of the smallest common denominator? The organization might benefit from the application of the findings of this study in the creation of artefacts to mitigate the challenges presented herein. Development, testing and implementation of such a remedy might form a very interesting experience as an extension to the presented work.

The typology of workarounds discovered through this study would certainly benefit from expansion and arranging. In the same way, the characteristics of workarounds created will need rigid testing and structuring so that they can be useful for the interpretation of situated use.

In closing their article, Schmidt and Wagner (2004) asks how to “*reduce the cost and increase the reliability of the distributed cooperative processes of producing and maintaining classification systems, notations, nomenclatures, procedures, etc.*?” (Schmidt and Wagner 2004, 402). In this

thesis, we have seen how some bricolages, as their own systems for ordering and structuring, are implemented and how they can take over for the formal system in many of its intents and purposes. Could the proliferation of these bricolages within a large company be beneficial to the organization? Could there be efficient ways of exchanging and improving each other's bricolages, to exploit all the work that has been undertaken – repeatedly – by different departments in the organization? In short, could there be an app store for workarounds and bricolages?

References

- Barley, Stephen R. 1986. "Technology as an occasion for structuring: Evidence from observations of CT scanners and the social order of radiology departments." *Administrative science quarterly*:78-108.
- Barley, Stephen R, and Julian E Orr. 1997. "Introduction: the neglected workforce." *Between craft and science: Technical work in US settings*:1-19.
- Baumer, Eric PS, Mark Sueyoshi, and Bill Tomlinson. 2011. "Bloggers and Readers Blogging Together: Collaborative Co-creation of Political Blogs." *Computer Supported Cooperative Work (CSCW)* no. 20 (1):1-36.
- Berg, Marc. 1997. On Distribution, Drift and the Electronic Medical Record: Some Tools for a Sociology of the Formal. Paper read at The Fifth European Conference on Computer Supported Cooperative Work.
- Bowers, John, Graham Button, and Wes Sharrock. 1995. Workflow from within and without. Paper read at Proceedings of the Fourth European Conference on Computer-Supported Cooperative Work ECSCW'95.
- Bratteteig, Tone. 2004. "Making change." *Dealing with relations between design and use. Dr. Philos dissertation, Department of Informatics, University of Oslo.*
- Checkland, Peter, and Sue Holwell. 1998. "Action research: its nature and validity." *Systemic Practice and Action Research* no. 11 (1):9-21.
- Ciborra, Claudio. 2000. *From control to drift: the dynamics of corporate information infrastructures*: Oxford University Press on Demand.
- Ciborra, Claudio U. 1997. "Introduction: what does groupware mean for the organizations hosting it?" In *Groupware and teamwork*, 1-19. John Wiley & Sons, Inc.
- Ciborra, Claudio U., and Gerardo Patriotta. 1996. "Groupware and teamwork in new product development: the case of a consumer goods multinational." In *Groupware and teamwork*, 121-142. John Wiley & Sons, Inc.
- Cooley, Charles Horton. 1964. *Human nature and the social order*: Transaction Publishers.
- Coover, Michael D, and Lori Foster Thompson. 2001. *Computer supported cooperative work: Issues and implications for workers, organizations, and human resource management*: Sage Publications, Inc.

References

- Corbin, J.M., and A. Strauss. 1990. "Grounded theory research: Procedures, canons, and evaluative criteria." *Qualitative sociology* no. 13 (1):3-21.
- Crang, Mike, and Ian Cook. 2007. *Doing ethnographies*: SAGE Publications Limited.
- Gasser, Les. 1986. "The integration of computing and routine work." *ACM Transactions on Information Systems (TOIS)* no. 4 (3):205-225.
- Grabot, Bernard, Anne Mayère, and Isabelle Bazet. 2008. *ERP systems and organisational change: a socio-technical insight*: Springer.
- Grudin, Jonathan. 1988. Why CSCW applications fail: problems in the design and evaluation of organizational interfaces. Paper read at Proceedings of the 1988 ACM conference on Computer-supported cooperative work.
- . 1994. "Groupware and social dynamics: eight challenges for developers." *Communications of the ACM* no. 37 (1):92-105.
- Hanseth, Ole, and Kristin Braa. 1998. Technology as traitor: Emergent SAP infrastructure in a global organization. Paper read at Proceedings of the international conference on Information systems.
- Kallinikos, Jannis. 2004. "Deconstructing information packages: Organizational and behavioural implications of ERP systems." *Information technology & people* no. 17 (1):8-30.
- Kleinman, Daniel Lee, and Steven P Vallas. 2001. "Science, capitalism, and the rise of the "knowledge worker": The changing structure of knowledge production in the United States." *Theory and Society* no. 30 (4):451-492.
- Lévi-Strauss, Claude. 1966. *The savage mind*. London: Weidenfeld and Nicolson.
- Orlikowski, Wanda J. 1993. "CASE tools as organizational change: investigating incremental and radical changes in systems development." *MIS Q.* no. 17 (3):309-340. doi: 10.2307/249774.
- . 1996. "Evolving the notes: organizational change around groupware technology." In *Groupware and teamwork*, 23-59. John Wiley & Sons, Inc.
- Orlikowski, Wanda Janina. 1995. *Evolving with Notes: Organizational change around groupware technology*: Sloan School of Management, Massachusetts Institute of Technology.
- Ribes, David, and Charlotte P Lee. 2010. "Sociotechnical studies of cyberinfrastructure and e-Research: Current Themes and Future Trajectories." *Computer Supported Cooperative Work (CSCW)* no. 19 (3):231-244.
- Rodden, Tom, and Gordon Blair. 1991. CSCW and distributed systems: The problem of control. Paper read at Proceedings of the Second European Conference on Computer-Supported Cooperative Work ECSCW'91.

- Schmidt, Kjeld. 1999. "Of maps and scripts: The status of formal constructs in cooperative work." *Information and software technology* no. 41 (6):319-329.
- . 2002. "The Problem with Awareness: Introductory Remarks on Awareness in CSCW." *Computer Supported Cooperative Work (CSCW)* no. 11 (3):285-298.
- Schmidt, Kjeld, and Liam J. Bannon. 1992. "Taking CSCW seriously." *Computer Supported Cooperative Work (CSCW)* no. 1 (1):7-40.
- Schmidt, Kjeld, and Ina Wagner. 2004. "Ordering systems: Coordinative practices and artifacts in architectural design and planning." *Computer Supported Cooperative Work (CSCW)* no. 13 (5):349-408.
- Silverman, David. 1998. "Qualitative research: meanings or practices?" *Information Systems Journal* no. 8 (1):3-20.
- Star, Susan Leigh. 2010. "This is not a boundary object: Reflections on the origin of a concept." *Science, Technology & Human Values* no. 35 (5):601-617.
- Star, Susan Leigh, and Karen Ruhleder. 1994. Steps towards an ecology of infrastructure: complex problems in design and access for large-scale collaborative systems. Paper read at Proceedings of the 1994 ACM conference on Computer supported cooperative work.
- Strauss, Anselm L. 1987. *Qualitative analysis for social scientists*: Cambridge University Press.
- Suchman, Lucy A. 2007. *Human-machine reconfigurations: Plans and situated actions*: Cambridge University Press.
- Thomas, Robert Joseph. 1994. *What machines can't do: Politics and technology in the industrial enterprise*: University of California Press.
- Wynn, Eleanor. 1997. "Groupware in a regional health insurer: local innovations and formative context in transition." In *Groupware and teamwork*, 211-236. John Wiley & Sons, Inc.
- Zuboff, Shoshana. 1985. "Automate/informate: the two faces of intelligent technology." *Organizational Dynamics* no. 14 (2):5-18.
- . 1988. *In the age of the smart machine: The future of work and power*: Basic Books.

List of Figures

Figure 2.1	Phases of grounded theory as applied in this study. The data was collected through semi-structured interviews.	10
Figure 2.2	An example of codes applied to an interview excerpt. Same data as Table 2.2.	17
Figure 2.3	The codes developed after analyzing interviews 7 and 8, as clustered by the application NVivo. A subselection is sorely needed. The dendrogram has been split in two to facilitate the page layout.	23
Figure 2.4	A random selection of codes for analysis yields a very straightforwardly structured dendrogram.	23
Figure 2.5	An example of a more critically selected subset of codes for analysis. A refined selection of codes for similarity analysis yields a much more interpretable and interesting result.	24
Figure 2.6	A proposed interpretation of the relationship between some of the observed effects: An integrative diagram highlighting the relationships between the core categories discovered in this analysis.	27
Figure 3.1	“Purchase to pay”, the acquisition process at the company, as described in internal workflow description documentation.	34
Figure 3.2	The purchasing process, following the main process, as performed by Ben. Note that the approval from the accounting department is never communicated back to Ben in any way, save for the eventual discovery that the invoice would never get paid.	41
Figure 3.3	The purchasing process, post-hoc, as performed by Dina. The technician seemingly has a carte blanche to perform purchases as he sees fit without notifying Dina.	43
Figure 3.4	Hand drawing of Fabian’s internally developed spreadsheet to keep track of purchases and their related activities. The spreadsheet was praised because it offered flexibility while imposing few restrictions on data input. The spreadsheet does not process the data in any way, it only structures and stores the data.	44
Figure 4.1	Portion of workday reported spent in the ERP system vs whether the users knew of workarounds.	84
Figure 4.2	Perception of adequate access in ERP system vs prevalence of workarounds.	85
Figure 4.3	Perception of adequate training in ERP system vs prevalence of workarounds.	86
Figure 4.4	Perception of adequate training in ERP system vs cause of workarounds.	88

List of Figures

Figure 6.1	Positioning the situated use as workarounds and bricolages reveals that the latter is more beneficial than the former. Size of the bubbles roughly represent their regularity in the study.	102
Figure 6.2	The reinforcing feedback loop of peripherality.....	105

List of Tables

Table 2.1	Different sources and their applicability in grounded theory analysis.	13
Table 2.2	An example of theory extraction from an interview excerpt. Same data as Figure 2.2.	17
Table 2.3	Number of codes assigned to some of the interviews, and the number of times these codes were applied during the text.....	20
Table 2.4	The 20 most frequently used codes along with the number of times they were used across interviews 7 and 8, sorted by sum descending, shortly after the first few rounds of open coding were completed.	21
Table 2.5	A summary of highly subjective pros et contras for grounded theory methodology.....	28
Table 3.1	Types or categories of workarounds and bricolages observed in the study, and their characteristics.	74
Table 3.2	Identified characteristics of workarounds and bricolages in situated use.	76
Table 4.1	Self-reported cause of workarounds among users who reported knowing of workarounds.	83
Table 6.1	Types or categories of workarounds and bricolages observed in the study, and their characteristics. Also shown in Table 3.1.	101
Table 6.2	Identified characteristics of workarounds and bricolages in situated use. Also shown in Table 3.2.	104

Attachments

1. Interview Guide for Semi-Structured Interviews, Fall 2012
2. Opinion Poll Questionnaire
3. Data for diagrams in chapter 4

Attachment 1:

Guide for Semi-Structured Interviews, Fall 2012

- What is your job?
 - How long have you been doing it?
 - What are you doing?
- Explain to me the purchasing processes in the organization.
- How do you use the ERP systems?
 - Is this how you desire to work?
 - Is this the best way to work?
- How is work coordinated in your department? How do you receive purchase requests?
- Is the system working for you? For you department?
 - Whose system is it?
- Is the system restrictive? Why?
- Are you faithful to the system?
 - Do you have alternative systems/solutions? Why?
 - Do you think all the departments in the organization can use the same system, or are different adaptations required?
- Explain what the data you are punching is used for.
- Is there room for feedback or improvement suggestions?
 - Have you ever provided any feedback?
 - How could the system be better?
- Do you desire another system?

Attachment 2:

Opinion Poll Questionnaire

The WEF is the web-based ERP-system front-end.

Question	Response options
Which division of [the organization] do you work for?	<ul style="list-style-type: none"> • [List of units, approx. 20]
What is the primary nature of your work?	<ul style="list-style-type: none"> • Supplies • IT • [Redacted] • Personell • Economy • Maintenance • Other
Are you a manager?	<ul style="list-style-type: none"> • Y/N
1. How would you rate your knowledge of WEF?	<ul style="list-style-type: none"> • Knows very well • Knows well • Knows a bit • No knowledge
2. What do you associate with WEF?	<ul style="list-style-type: none"> • Administrative system • [Redacted] • Management system • Other • Don't know
3. WEF is useful for [the organization] as a whole	<ul style="list-style-type: none"> • Totally agree • Agree • Agree a bit • Disagree a bit • Disagree • Totally disagree
4. WEF is a useful tool for me in my workday	<ul style="list-style-type: none"> • Totally agree • Agree • Agree a bit • Disagree a bit • Disagree • Totally disagree
5. I experience that today's solution in WEF contributes to freeing resources for more [primary business product]	<ul style="list-style-type: none"> • Totally agree • Agree • Agree a bit • Disagree a bit • Disagree • Totally disagree
6. How much time of your workday do you use [ERP system]?	<ul style="list-style-type: none"> • Less than 25% • 25% – 50% • 50% – 75% • More than 75%
7. Do you, or do you know anybody else, that has developed alternative ways to work, where one should really use [ERP system]?	<ul style="list-style-type: none"> • Y/N
What is the reason for this?	<ul style="list-style-type: none"> • Already had working processes • Did not wish to change established methods of working • [ERP system] doesn't cover our needs • [ERP system] has a too high threshold of use/to bad user interface • Other/don't know

Question	Response options
8. I feel that I have the necessary rights in [ERP system] to perform my job	<ul style="list-style-type: none"> • Totally agree • Agree • Agree a bit • Disagree a bit • Disagree • Totally disagree
9. I feel that I have the necessary education in [ERP system] to perform my job.	<ul style="list-style-type: none"> • Totally agree • Agree • Agree a bit • Disagree a bit • Disagree • Totally disagree
10. How would you rate your knowledge of project HRM in WEF?	<ul style="list-style-type: none"> • No knowledge • Knows it a bit • Knows it well • Knows it very well
11. How would you rate your knowledge of [development project]?	<ul style="list-style-type: none"> • No knowledge • Knows it a bit • Knows it well • Knows it very well
12. I have positive expectations of future solutions in WEF	<ul style="list-style-type: none"> • Totally agree • Agree • Agree a bit • Disagree a bit • Disagree • Totally disagree
13a. I expect improved, holistic work processes as a result of new projects in WEF	<ul style="list-style-type: none"> • Totally agree • Agree • Agree a bit • Disagree a bit • Disagree • Totally disagree
13b. I expect improved technology as a result of new projects in WEF	<ul style="list-style-type: none"> • Totally agree • Agree • Agree a bit • Disagree a bit • Disagree • Totally disagree
13c. I expect improved organization as a result of new projects in WEF	<ul style="list-style-type: none"> • Totally agree • Agree • Agree a bit • Disagree a bit • Disagree • Totally disagree
14. I expect that future solutions in WEF will free resources for more [primary business product]	<ul style="list-style-type: none"> • Totally agree • Agree • Agree a bit • Disagree a bit • Disagree • Totally disagree
15. I expect future solutions in WEF to make my workday easier, or contribute to me doing my tasks with a higher degree of quality	<ul style="list-style-type: none"> • Totally agree • Agree • Agree a bit • Disagree a bit • Disagree • Totally disagree

Attachment 3:

Data for diagrams in chapter 4

Table A3.1 Portion of workday reported spent in the ERP system vs whether the users knew of workarounds.

Portion of workday spent in the ERP system	Uses workarounds	Does not use workarounds
Less than 25%	34 %	66 %
25% - 50 %	43 %	57 %
50 % - 75 %	47 %	53 %
More than 75%	40 %	60 %

Table A3.2 Perception of adequate access in ERP system vs workarounds.

Has adequate access	All	Uses workarounds	Does not use workarounds
Totally agree	15 %	26 %	74 %
Agree	40 %	33 %	67 %
Agree a bit	19 %	36 %	64 %
Disagree a bit	12 %	44 %	56 %
Disagree	8 %	47 %	53 %
Totally disagree	6 %	46 %	54 %
Totals	100 %	36 %	64 %

Table A3.3 Perception of adequate training in ERP system vs prevalence of workarounds.

Has adequate training	All	Uses workarounds	Does not use workarounds
Totally agree	7 %	22 %	78 %
Agree	25 %	30 %	70 %
Agree a bit	22 %	36 %	64 %
Disagree a bit	20 %	36 %	64 %
Disagree	15 %	44 %	56 %
Totally disagree	11 %	47 %	53 %
Totals	100 %	36 %	64 %

Table A3.4 Perception of adequate training in ERP system vs cause of workarounds.

	Totally agree	Agree	Agree a bit	Disagree a bit	Disagree	Totally disagree
Already had working processes	7 %	19 %	19 %	30 %	9 %	17 %
Did not wish to change established methods of working	11 %	44 %	22 %	11 %	7 %	5 %
[ERP system] doesn't cover our needs	4 %	23 %	25 %	21 %	16 %	11 %
[ERP system] has a too high threshold of use/ too bad user interface	3 %	13 %	22 %	20 %	24 %	19 %
Other/don't know	7 %	32 %	18 %	15 %	17 %	10 %
(No workarounds)	9 %	27 %	22 %	20 %	13 %	10 %
Totals	7 %	25 %	22 %	20 %	15 %	11 %